



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF CONTROL AND INSTRUMENTATION

## ZÍSKÁVÁNÍ DAT Z TABLETU

TABLET DATA ACQUISITION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MARTIN ZVAK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. MILOSLAV RICHTER, Ph.D.

BRNO 2010



**VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky  
a komunikačních technologií**

**Ústav automatizace a měřicí techniky**

# Diplomová práce

magisterský navazující studijní obor  
**Kybernetika, automatizace a měření**

**Student:** Bc. Martin Zvak

**ID:** 78380

**Ročník:** 2

**Akademický rok:** 2009/2010

**NÁZEV TÉMATU:**

**Získávání dat z tabletu**

## POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s problematikou vstupních zařízení typu tablet. Navrhněte tvary vstupních dat a vytvořte testovací vzory a testovací databáze vhodné pro zpracování těchto dat. Navrhněte základní metody pro zpracování dat a na základě realizace stanovte vhodnost zvolených příznaků pro vyhodnocení. Na základě návrhu napište program pro získávání a zpracování dat z tabletu..

## DOPORUČENÁ LITERATURA:

[www.pointing.com](http://www.pointing.com)

Žára J., Beneš B., Sochor J., Felkel P.: Moderní počítačová grafika, Computer Press, 1998, ISBN 80-251-0454-0

Hlaváč V., Šonka M.: Počítačové vidění, Grada, Praha 1992, ISBN 80-85424-67-3

Faugeras O.: Three-Dimensional Computer Vision, The MIT Press 1993

**Termín zadání:** 8.2.2010

**Termín odevzdání:** 24.5.2010

**Vedoucí práce:** Ing. Miloslav Richter, Ph.D.

**prof. Ing. Pavel Jura, CSc.**  
*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## Abstrakt

Spolehlivá autorizácia a overovanie identity sa v súčasnosti stávajú nevyhnutnými pre mnohé každodenné činnosti akými sú nastupovanie do lietadla, prevádzanie finančných operácií apod.

Táto práca pojednáva o možnostiach návrhu vhodnej metódy získavania, spracovania a vyhodnotenia dát z tabletu ako takého, so zameraním na možnosť získavať dáta z grafického tabletu použitého v spojení s osobným počítačom, pre účely overenia identity. Obsahuje prehľad biometrie, ako samostante sa rozvíjajúcej vednej disciplíny v priebehu posledných desaťročí 20. storočia až po súčasnosť, súhrn vývoja grafických tabletov, aktuálny stav v súčasnosti, stručný prehľad najrozšírenejších technológií využívaných pri ich konštrukcii a samotné rozdelenie. Ďalej sa zameriava na návrh a popis implementácie vhodného postupu získavania dát z poskytnutého tabletu Trust TB - 7300, s využitím dostupného vývojového balíka pre Tablet PC spoločnosti Microsoft, a porovnanie modelu WinForms a WPF (.NET Framework) za súčasného deklarovania vhodnosti použitia práve druhého z nich. V neposlednom rade je, na základe odozvy od dostupnej množiny užívateľov, kladený dôraz na návrh vhodných typov vstupných dát, testovacích vzorov, vhodnej metódy a aplikácie pre zber a spracovanie dát získaných z grafického tabletu, ako aj ich použitie pre proces overenia identity daného jedinca. V konečnom dôsledku je tak možné previesť zhodnotenie presnosti danej metódy na množine užívateľov, jej vhodnosti pre použitie a nedostatkov.

Na základe analýzy danej problematiky je možné navrhnutú metódu prehlásiť za nedostatočne optimálnu – je totiž bezpečná, ale nepresná s malou mierou stálosti a opakovateľnosti. Jedná sa však len o prvotný návrh, pričom použitie v aplikáciách každodenného života je zatiaľ nemožné a proces prechodu z laboratórnych podmienok by si vyžiadal ďalšie pokračovanie v testovaní a zdokonaľovaní postupu.

## Kľúčové slová

biometria, získavanie dát z tabletu, grafický tablet, C#, WinForms, WPF, RealTimeStylus

## Abstract

Reliable authorization and authentication is now becoming necessary for many everyday activities such as boarding an aircraft, doing financial transactions etc.

This work deals with the possibility of proposing an appropriate method of acquisition, processing and evaluation of data from the tablet itself, focusing on the possibility of obtaining data from the graphics tablet used in conjunction with a personal computer, for the purpose of verifying identity. It contains an overview of biometrics, as independent developing science over the last decades of the 20th century to the present, a summary of the development of a graphics tablet, the current situation, a brief overview of the most popular technologies used in their construction and distribution. Furthermore, it focuses on the design and implementation of appropriate procedures to obtain data provided by the Trust tablet TB - 7300, using available software development kit for the Microsoft Tablet PC, and a comparison of WinForms and WPF model (.NET Framework), with the declaration of suitability for the use of the second one. Last but not least, according to the response from the available set of users, this work focuses on the design of appropriate types of input data, test patterns, method and application for collecting and processing data from the graphic tablet, as well as their use for the verification of the identity of the individual. Ultimately, it is possible to discuss the accuracy of the method on a set of users, its suitability for use and shortcomings.

Based on the analysis of the issue, the proposed method can be declared as under-optimal - it is safe, but inaccurate with low stability and repeatability. It is also important to accent that this is just the initial proposal, the use of applications in everyday life is for current phase impossible and the transition from laboratory conditions would require further development and improvement of the process.

## Keywords

biometrics, tablet data acquisition, graphic tablet, C#, WinForms, WPF, RealTimeStylus

## Bibliografická citácia

ZVAK, M. *Získávání dat z tabletu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 115 s. Vedoucí diplomové práce Ing. Miloslav Richter, Ph.D.

## Prehlásenie

„Prehlasujem, že svoju bakalársku prácu na tému "Systém optickej identifikácie osôb" som vypracoval samostatne pod vedením vedúceho bakalárskej práce a s použitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej bakalárskej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto bakalárskej práce som neporušil autorské práva tretích osôb, hlavne som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a som si plne vedomý následkov porušenia ustanovení § 11 a nasledujúceho autorského zákona č. 121/2000 Sb., vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovení § 152 trestného zákona č. 140/1961 Sb.“

V Brne dňa : 24.mája 2010

Podpis:

## Pod'akovanie

Ďakujem týmto Ing. Miloslavovi Richterovi, Ph.D za cenné pripomienky a rady pri vypracovaní diplomovej práce.

V Brne dňa : 24.mája 2010

Podpis:

## Obsah

<b>1. ÚVOD .....</b>	<b>12</b>
<b>2. BIOMETRIA.....</b>	<b>13</b>
2.1 Biometria a minulosť .....	13
2.2 Biometria a prítomnosť .....	15
2.2.1 Aplikácia biometrie v bežnom živote .....	17
2.2.2 Prehľad biometrických metód .....	18
2.2.3 Vzájomné porovnanie biometrických metód .....	22
<b>3. TABLET .....</b>	<b>24</b>
3.1 Čo je grafický tablet ? .....	24
3.2 Zaradenie .....	25
3.3 Historický vývoj .....	28
3.4 Súčasnosť .....	32
3.4.1 Parametre.....	32
3.4.2 Rozdelenie .....	34
3.4.3 Použitie.....	36
3.4.4 Predchádzanie vzniku ochorení.....	40
<b>4. ZÍSKAVANIE DÁT Z GRAFICKÉHO TABLETU.....</b>	<b>41</b>
4.1 Tablet PC.....	41
4.1.1 SDK .....	41
4.2 Použitý grafický tablet .....	42
4.3 .NET Framework .....	43
4.3.1 WinForms.....	46
4.3.2 Windows Presentation Foundation (WPF).....	47
4.4 Prístup a manipulácia so vstupnými dátami .....	48
4.4.1 Získanie a reprezentácie vstupných dát .....	50
4.4.2 Správa udalostí .....	52
4.4.3 Dáta zo stylusu .....	53
4.5 Model WinForms (varianta spravovaný kód).....	54
4.5.1 Architektúra StylusInput API.....	55
4.5.2 Špecifikácia pluginov – synchronný a asynchronný .....	55

4.5.3	Dátový tok.....	56
4.5.4	„HIMETRIC“ súradný systém .....	58
4.6	Model WinForms - realizácia navrhnutého riešenia.....	60
4.7	Model WPF .....	69
4.7.1	Architektúra StylusPlugIns API .....	69
4.7.2	Dátový tok.....	70
4.7.3	Vyobrazenie digitálneho atramentu (rendering).....	72
4.7.4	Kolekcia pluginov (StylusPluginCollection) .....	73
4.7.5	„HIMETRIC“ súradný systém .....	74
4.8	Model WPF - realizácia navrhnutého riešenia .....	74
<b>5.</b>	<b>REALIZÁCIA NAVRHNUTÉHO PRÍSTUPU .....</b>	<b>75</b>
5.1	Vedľajšie časti aplikácie.....	79
5.1.1	Testovanie prítomnosti (pripojenia) grafického tabletu.....	79
5.1.2	Demonštračná časť aplikácie.....	81
5.2	Hlavné časti aplikácie .....	84
5.2.1	Návrh vhodných typov vstupných dát .....	85
5.2.2	Prihlásenie do aplikácie a uchovávanie zvolených údajov .....	90
5.2.3	Prevedenie kalibrácie pre zásobník testovacích vzorov.....	92
5.2.4	Zber dát pri použití navrhnutých vzorov a ich následné spracovanie .....	94
5.2.5	Testovanie - overenie identity užívateľa.....	99
5.2.6	Zhodnotenie.....	106
<b>6.</b>	<b>ZÁVER .....</b>	<b>110</b>
<b>7.</b>	<b>POUŽITÁ LITERATÚRA.....</b>	<b>112</b>



## Zoznam obrázkov

Obrázok č. 1: Delenie biometrických metód.....	16
Obrázok č. 2: Nasadenie biometrických zariadení spoločnosťou Walt Disney [7]...	17
Obrázok č. 3: Príklady aplikácie biometrie v bežnom živote [7] .....	18
Obrázok č. 4: 4 základné vzory používané pre klasifikáciu [8] .....	19
Obrázok č. 5: Štandardná forma „eigenfaces“ a ich použitie [9] .....	19
Obrázok č. 6: Systém pre snímanie geometrie ľudskej ruky.....	20
Obrázok č. 7: Oblasť dúhovky „rozbalená“ do normalizovanej šablóny [10] .....	20
Obrázok č. 8: Profily DNA [11] .....	21
Obrázok č. 9: (a) Rozpoznávanie dynamiky vlastnoručného podpisu, (b) Sledovanie dynamiky klávesových úderov [12].....	21
Obrázok č. 10: Grafický tablet Intuos3 (Wacom) [14] .....	24
Obrázok č. 11: Myš Razer-Diamondback3G .....	25
Obrázok č. 12: Mini-myš Belkin (vľavo), Trackball Microsoft (vpravo).....	26
Obrázok č. 13: „Touchpad“ notebooku HP .....	26
Obrázok č. 14: Grafický tablet Bamboo (Wacom) [14] (vľavo), dotyková obrazovka (vpravo) .....	27
Obrázok č. 15: Pákový ovládač Cyber Knight (Rockfire) (vľavo), analógový ovládač (vpravo) .....	27
Obrázok č. 16: Umiestnenie uvedeného typu ovládača .....	28
Obrázok č. 17: Model teleautografu z roku 1893 [17].....	28
Obrázok č. 18: Pôvodný a prenesený text (teleautograf) [17].....	29
Obrázok č. 19: RAND tablet (Grafacon) [18] .....	30
Obrázok č. 20: Prospekty a samotný KoalaPad.....	30
Obrázok č. 21: Summagrid na dobovej fotografii [20] (vľavo), Bit-Pad One na dobovej fotografii [20] (vpravo).....	31
Obrázok č. 22: Ponuka grafického editoru KoalaPainter [19] .....	36
Obrázok č. 23: Grafické rozhranie IME.....	37
Obrázok č. 24: Grafické rozhranie programu EverNote .....	38
Obrázok č. 25: SignDoc spoločnosti Softpro (vľavo), Bio-Pen spoločnosti DBS, Inc. (vpravo) .....	39

Obrázok č. 26: Dielo mladej rumunskej umelkyne (použitie grafického tabletu) (vľavo), Komiks AppleGeeks (vpravo) .....	40
Obrázok č. 27: Pohľad na usporiadanie klasickej kancelárie.....	40
Obrázok č. 28: Pohľad na použitý grafický tablet .....	42
Obrázok č. 29: Prehľad verzií frameworku, spolu s komponentami [24].....	44
Obrázok č. 30: Prehľad štruktúry pre .NET Framework verzie 3.0 [28].....	47
Obrázok č. 31: Model WinForms. ....	51
Obrázok č. 32: Model WPF.....	52
Obrázok č. 33: Porovnanie modelu tried pre správu udalostí, ktoré môžu byť vyvolané pri použití stylusu [30] .....	53
Obrázok č. 34: Porovnanie modelu tried pre zachytenie dát zo stylusu [30].....	54
Obrázok č. 35: Znázornenie toku dát cez RealTimeStylus [31].....	57
Obrázok č. 36: Súradný priestor s počiatkom v bode (0,0).....	58
Obrázok č. 37: Súradný priestor modifikovaný spustenou aplikáciou .....	59
Obrázok č. 38: Štart aplikácie bez pripojeného grafického tabletu.....	62
Obrázok č. 39: Vizualizácia získaných dát na definovanej oblasti .....	67
Obrázok č. 40: Prekročenie hranice definovanej oblasti.....	68
Obrázok č. 41: Indexovanie vytvorených „strokes“ .....	68
Obrázok č. 42: Znázornenie modelu využitia vlákien pri zadávaní vstupných dát užívateľom, teda tvorbe „stroke“ [32] .....	70
Obrázok č. 43: Znázornenie procesu vyobrazenia digitálneho atramentu [32].....	72
Obrázok č. 44: Znázornenie možného usporiadania pluginov v kolekcii [32] .....	73
Obrázok č. 45: Hodnoty korelačného koef. pre rôzne situácie [34] .....	78
Obrázok č. 46: Navigačný panel aplikácie – grafický tablet nie je pripojený .....	80
Obrázok č. 47: Navigačný panel aplikácie – grafický tablet je pripojený .....	80
Obrázok č. 48: Okno demonštračnej časti aplikácie.....	83
Obrázok č. 49: Diagram prevedenia algoritmu navrhutej metódy .....	84
Obrázok č. 50: Písmeno L na pracovnej ploche aplikácie (model WinForms).....	86
Obrázok č. 51: Písmeno L na pracovnej ploche aplikácie (model WPF) .....	87
Obrázok č. 52: Prehľad charakteristík, ktoré boli pri návrhu vzorov predmetom záujmu (zľava ostrý, pravý, tupý uhol a hladký priebeh) .....	88

Obrázok č. 53: Prehľad navrhnutých vzorov .....	89
Obrázok č. 54: Finálny zásobník navrhnutých testovacích vzorov .....	90
Obrázok č. 55: Náhľad záznamu uchovávaných údajov pre prihláseného užívateľa.91	
Obrázok č. 56: Kaliber použitý v procese kalibrácie.....	93
Obrázok č. 57: Zásobník s návrhom testovacích vzorov + kaliber .....	94
Obrázok č. 58: Náhľad okna časti aplikácie pre kalibráciu .....	94
Obrázok č. 59: Body definované na obvode časti vzoru a vytvárajúce štandardnú šablónu; zvýraznená tolerancia okolia jedného bodu .....	95
Obrázok č. 60: Výber oblastí pre vzor štvorec; 3 oblasti záujmu; posledná slúži ako štartovacia a finalizačná .....	96
Obrázok č. 61: Jednotlivé varianty umiestnenia štartovacej / finalizačnej pozície pre vzor štvorec, spolu so smerom obkresľovania .....	96
Obrázok č. 62: Oblasť kladnej (červená farba) a zápornej (modrá farba) diferencie.97	

## **Zoznam tabuliek**

Tabuľka č. 1: Porovnanie biometrických charakteristík, podľa A. K. Jain.....	23
Tabuľka č. 2: Prehľad dostupných verzií .NET Framework.....	44
Tabuľka č. 3: Význam korelačných koeficientov podľa J. Cohena.....	79
Tabuľka č. 4: Porovnanie použitej charakteristiky s najpodobnejšou z doplnkových biometrických charakteristík .....	108

## 1. ÚVOD

Hlavným cieľom tejto práce je návrh vhodnej metódy získavania, spracovania a vyhodnotenia dát z tabletu ako takého, so zameraním na možnosť získavať dáta z grafického tabletu použitého v spojení s osobným počítačom.

Nasledujúce kapitoly 2 a 3 sú teoretickým úvodom k danej problematike. Je kladený dôraz na priblíženie biometrie, ako samostante sa rozvíjajúcej vednej disciplíny v priebehu posledných desaťročí 20. storočia až po súčasnosť, spolu s dostupnými biometrickými metódami. Ďalej sa jedná o popis grafického tabletu ako jedného z nespočetného množstva polohovacích zariadení, historického vývoja, aktuálneho stavu v súčasnosti, stručného prehľadu najrozšírenejších technológií využívaných pri konštrukcii a samotného rozdelenia. Následne je značná pozornosť venovaná použitiu tabletu a to napr. pre účely rozšírenia možností interakcie užívateľa s osobným počítačom, identifikácie jedinca, tvorby umenia apod.

Štvrtá kapitola prináša popis dvoch variant získavania dát z grafického tabletu, jedná sa o tzv. WinForms a WPF model (zo softvérového vývojového balíka spoločnosti Microsoft, v prostredí operačného systému Windows), špecifikáciu použitého rozhrania, možnosti zachytávania a následného spracovania dátového toku až po konečnú archiváciu a vizualizáciu získaných dát. Zároveň je poskytnutý detailný pohľad na navrhnuté aplikácie, využívajúce popísané riešenia a v konečnom dôsledku prevedený výber prístupu vhodného pre ďalšie pokračovanie, ktorým sa v danom prípade stal prístup založený na modeli WPF.

V poradí piata kapitola už nie je orientovaná len na všeobecné teoretické rozborov metód a ich postupov. Odhliadnuc od jej prvých častí, venovaných priblíženiu riešenia a štruktúry navrhovanej aplikácie, sa venuje detailnému postupu realizácie uvedenej metódy a jej jednotlivých krokov – od tvorby testovacej množiny užívateľov až po samotný proces testovania – overenia identity jedinca. Získané výsledky spolu so zaznamenanými problémami sú zhrnuté v poslednej časti označenej ako zhodnotenie.

Celkové dosiahnuté výsledky, ako aj výsledky analýzy jednotlivých krokov metódy a využitých postupov, sú stručne zhrnuté v závere.

## 2. BIOMETRIA

Táto kapitola je teoretickým úvodom do celej problematiky biometrie ako samostatnej vednej disciplíny a postupne sú v nej priblížené najrozšírenejšie biometrické metódy používané v dnešnom svete.

Pojem “biometria” vychádza z dvoch gréckych slov, zo slova “bios” - znamenajúceho (život) a slova “metron” – vo význame (miera). Dnešné automatizované biometrické systémy sa stali použiteľnými len v priebehu minulých desaťročí, ako jeden z dôsledkov významných pokrokov na poli počítačového spracovania. Mnohé z týchto nových automatizovaných techník však vychádzajú z myšlienok, ktoré boli pôvodne koncipované predstovkami či dokonca tisíckami rokov.

### 2.1 BIOMETRIA A MINULOSŤ

Jeden z najstarších a najzákladnejších príkladov charakteristiky použitej pre rozpoznávanie jedincov vrámci spoločenstva je ľudská tvár. Od zrodenia civilizácie ľudia používali tváre na rozpoznanie známych (príslušníci rodín) a neznámych (cudzinci) jedincov. Táto zdanlivo jednoduchá úloha sa stávala čoraz náročnejšou s rastúcou populáciou a s rozvojom vyhovujúcich metód cestovania, ktoré privádzali mnoho nových jedincov do niekdajších malých komunít. Pojatie rozpoznávania človek – voči – človeku („jedna osoba so znakom vykazujúcim unikátnosť vrámci spoločenstva“) je dnes možné nájsť v metódach biometrie správania sa, akými sú rozpoznávanie rečníka a držania tela. Jednotlivci využívajú tieto charakteristiky, podvedome a tak trochu nevedome, na rozpoznanie známych jedincov na báze zo dňa – na – deň. („toho človeka som v minulých dňoch videl ...“)

Ostatné charakteristiky boli v rámci histórie ľudstva použité ako oveľa formálnejšie prostriedky pre rozpoznávanie. Príklady sú uvedené v pokračovaní:

- V jaskyni s odhadovaným vekom 31.000 rokov sú steny vyzdobené maľbami o ktorých sa vedci domnievajú, že sú dielom prehistorického človeka ktorý tu žil. Okolie týchto malieb tvoria početné odtlačky rúk ktoré sú akýmsi podpisom ich tvorcu.
- Joao de Barros, španielsky objaviteľ a spisovateľ, zaznamenal používanie odtlačkov prstov čínskymi obchodníkmi ako formy potvrdenia obchodných transakcií. Čínsky rodičia taktiež používali odtlačky prstov a chodidiel na rozlíšenie svojich detí.

Do polovice 19. storočia, s prudkým rastom miest spôsobeným priemyselnou revolúciou a výnosnejším hospodárením, sa čoraz častejšie objavovala vážna potreba identifikovať osoby. Obchodníci a úrady boli neustále konfrontovaní s čoraz väčšou a mobilnejšou populáciou a už sa viac nemohli spoliehať na vlastné skúsenosti a lokálne znalosti. Ovplynení dielami Jeremyho Bethama a ďalších Prospechárskych mysliteľov, trestné orgány v tomto období začali kodifikovať svoje predstavy o súdnych procesoch a spravodlivosti ktoré vytrvali až do dnešných dní. Právny systém zaobchádzal s tými, ktorí sa dostali do rozporu so zákonom prvýkrát miernejšie ako s viacnásobne trestanými. To viedlo k vytvoreniu oficiálneho systému ktorý zaznamenával trestné činy spolu s totožnosťou páchatel'a.

Prvým z dvoch prístupov bol „Bertillonov systém“ pre meranie rozmanitých romerov ľudského tela, majúci pôvod vo Francúzsku. Tieto rozmery boli zapísané na karty, ktoré boli následne roztriedené podľa výšky, dĺžky ruky alebo niektorého z ďalších parametrov. Táto oblasť sa nazývala „antropometria“.

Ďalším krokom bolo oficiálne použitie odtlačkov prstov policajnými oddeleniami. Tento proces sa objavil v Južnej Afrike, Ázii, a Európe. Ku koncu 19. storočia bola vyvinutá metóda na zaznamenávanie odtlačkov prstov, ktorá poskytovala možnosť opätovného prístupu k záznamom ako tomu bolo v prípade „Bertillonovej metódy“, avšak tu sa jednalo o špecifickejšiu metriku – vzory odtlačkov prstov.

Prvý robustný systém tohto druhu bol vyvinutý v Indii „Azizulom Haqueom“ pre Edwarda Henryho, generálneho policajného inšpektora pre oblasť Bengal, India. Tento systém, nazvaný Henry System, a mnohé jeho obmeny sa neustále používajú pre klasifikáciu odtlačkov prstov i v dnešnom svete [6].

Skutočné biometrické systémy sa začali objavovať v druhej polovici 20. storočia, v spojení s výstupmi počítačových systémov. Rodiaca sa oblasť tak viedla k explózií aktivít v danom obore od roku 1990 a začala sa objavovať v každodenných aplikáciách na počiatku 21. storočia.

## 2.2 BIOMETRIA A PRÍTOMNOSŤ

Spôľahlivá autorizácia (authorization) a taktiež overovanie (authentication) sa v súčasnosti stávajú nevyhnutnými pre mnohé každodenné činnosti akými sú nastupovanie do lietadla, prevádzanie finančných operácií a pod. Autorizácia je takmer vždy zabezpečená pre jednotlivca, prípadne malú skupinu jednotlivcov, zatiaľ čo overovanie je vo veľkej miere nasadzované na rozsiahle početné skupiny. Tieto dve zdanlivo jednoduché operácie sa však stávajú náročnými v prípade požiadavku na automatickosť s vysokou presnosťou, malou pravdepodobnosťou prerušenia a spoľahlivosťou. Navyše, užívateľ by mal byť obťažovaný v čo najmenšej miere a nemal by byť schopný poprieť uskutočnenú transakciu či odmietnuť jej vykonanie, čím sa len zvyšuje obťažnosť prevedenia samotnej úlohy.

Rozpoznávanie jedincov je zásadnou aktivitou vo vnútri našej spoločnosti a kultúry, odkedy je pre mnohé aplikácie zaručenie identity a autentičnosti zainteresovaných ľudí nevyhnutnosťou.

„Biometria“ tak predstavuje univerzálny pojem možný použiť alternatívne na popísanie charakteristiky alebo procesu.

Použitie ako charakteristika:

1. Zmerateľná biologická (anatomická a fyziologická) char. a charakteristika správania sa použiteľná pre automatizované rozpoznávanie.

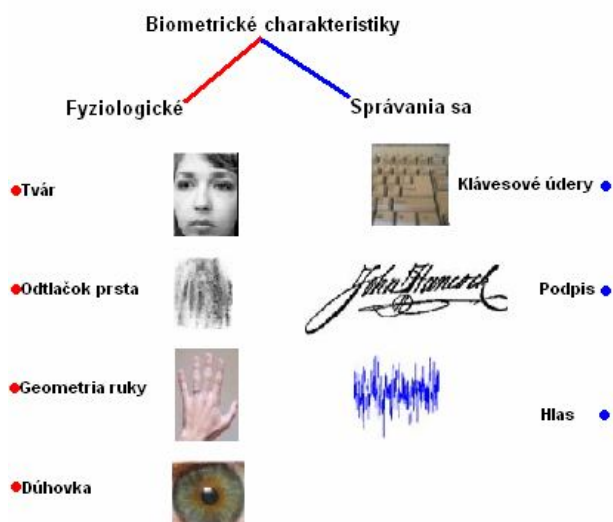
Použitie ako proces:

2. Automatizovaná metóda rozpoznávania jedinca založená na merateľných biologických (anatomických a fyziologických) char. a charakteristike správania sa.

V súčasnosti je tento pojem prezentovaný ako štúdium metód pre unikátne rozpoznávanie ľudí, založených na jednej, prípadne viacerých vrodených fyzických črtách alebo správaní sa jedinca. Hlavné biometrické charakteristiky možno rozdeliť do dvoch tried, ako znázorňuje Obrázok č. 1:

- fyziologické – súvisiace priamo s telom človeka ako takým. Najstaršou charakteristikou, používanou už viac ako 100 rokov, sú odtlačky prstov. Ďalšie príklady sú napr. rozpoznávanie tváre, geometria ruky, rozpoznávanie dúhovky atd.

- správania sa – spojené so správaním sa konkrétneho človeka. Typickým zástupcom tejto triedy je sledovanie dynamiky vlastnoručného podpisu. Ďalšie príklady sú napr. rozpoznávanie hlasu, klávesových úderov atd.



Obrázok č. 1: Delenie biometrických metód



Biometrické systémy schopné využiť spomínané charakteristiky boli vyvíjané a testované niekoľko desaťročí, avšak len nedávno vstúpili do verejného vedomia a to vďaka veľkoprofilovým aplikáciám, použitiu vo filmoch (napr. film The Island a často i pomerne nerealisticky, napr. film Minority Report) a zvýšenému využívaniu zo strany verejnosti pri každodenných aktivitách.

Mnohé spoločnosti pracujú na implementácii biometrických systémov pre zabezpečenie oblastí, udržiavanie časových záznamov a na zlepšenie pohodlia užívateľa. Napr., po mnohé roky kolos Disney World využíva biometrické zariadenia pre držiteľov sezónnych lístkov aby sa urýchlil a zjednodušil proces vstupu do jednotlivých parkov a zároveň sa zabezpečilo, aby bol lístok použitý len jedincom ktorému bol vydaný (Obrázok č. 2).



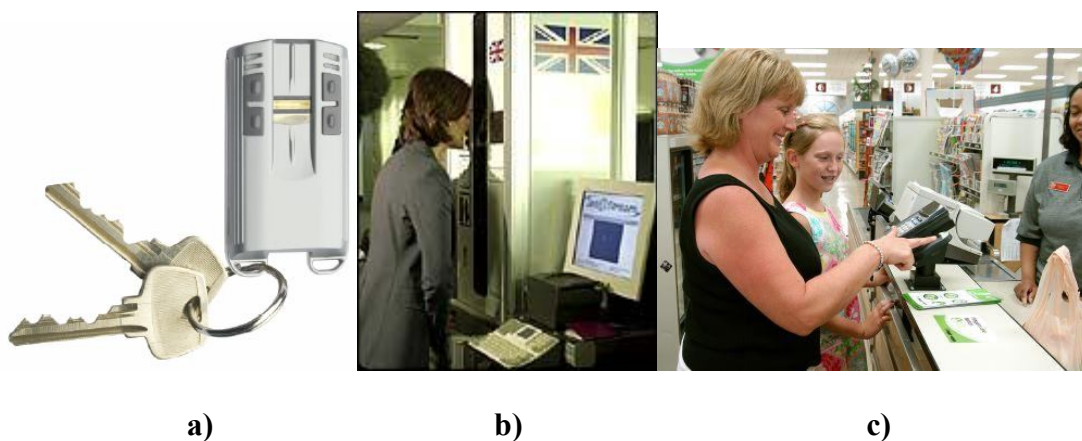
**Obrázok č. 2: Nasadenie biometrických zariadení spoločnosťou Walt Disney [7]**

### **2.2.1 Aplikácia biometrie v bežnom živote**

Zatiaľ čo biometrické systémy, obzvlášť systémy automatickej identifikácie odtlačkov prstov (napr. AFIS v USA), boli a sú využívané v súdnictve na identifikáciu osôb, pokroky na poli biometrických senzorov a porovnávacích algoritmov viedli k rozmiestneniu biometrického overovania v civilných i vládnych inštitúciách.

Biometria je tak používaná pre fyzickú kontrolu prístupu, počítačové prihlasovanie, hraničnú kontrolu a pod. Môže byť použitá na preverenie zákazníka počas transakcie prevedenej cez telefón a internet. V automobilovom priemysle nahrádza biometria bežné kľúče.

Ako dôsledok narastajúcich bezpečnostných hrozieb odsúhlasila organizácia ICAO (Medzinárodná Civilná Letecká Organizácia) použitie e-pasov (pasov so zapusteným čipom obsahujúcim snímky tváre držiteľa a ďalšie biometrické charakteristiky).



**Obrázok č. 3: Príklady aplikácie biometrie v bežnom živote [7]**

- a) integrované biometrické zariadenie (náhrada bežného kľúča)**
- b) letisko Heathrow (snímanie dúhovky)**
- c) pay-by-touch (platba dotykom)**

### **2.2.2 Prehľad biometrických metód**

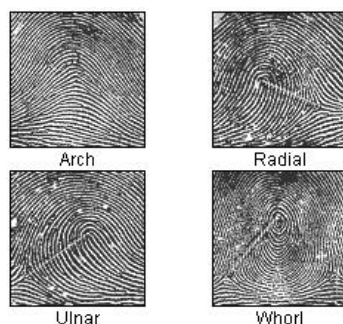
V súčasnosti dostupné biometrické metódy je možné rozdeliť do dvoch skupín z hľadiska intenzity použitia v aplikáciách každodenného života. Jedná sa tak o metódy označované ako bežne používané – rozpoznávanie odtlačkov prstov, tváre, rečníka, geometrie ruky/prsta, dúhovky – a metódy doplnkové – rozpoznávanie sietnice, analýza DNA, dynamika klávesových úderov, vlastnoručného podpisu a mnohé ďalšie.

Stručný popis metod, začínajúc s bežne používanými, je uvedený v pokračovaní.

#### *Rozpoznávanie odtlačkov prstov*

Manuálne porovnávanie odtlačkov prstov pre rozpoznávanie sa v praxi používalo po mnohé roky a stalo sa automatizovanou biometrickou identifikačnou technikou dvoch minulých desaťročí. Odtlačky prstov majú vzhľad nerovného povrch tvoreného „hrebeňmi“ (tmavé časti odtlačku) a „údoliami“ (biele nevyplnené miesta), ktoré tak formujú unikátnu šablónu pre každého jedinca.

Pri klasifikácii jednotlivých odtlačkov sa dnes celosvetovo používajú 4 základné vzory.

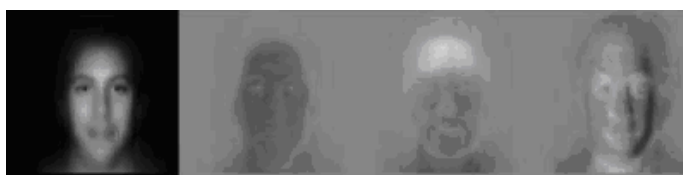


**Obrázok č. 4: 4 základné vzory používané pre klasifikáciu [8]**

#### *Rozpoznávanie tváre*

Človek je schopný rozpoznať tvár svojho blízkeho so značnou ľahkosťou, avšak v prípade rozpoznávania cudzincov dochádza k častým nepresnostiam. Od roku 1960 pracujú výskumné tímy v oblasti počítačového videnia na vývoji plne automatizovanej metódy pre rozpoznávanie jednotlivcov na základe črt tváre.

Prvou funkčnou metódou bolo použitie „eigenfaces“ („vlastných tvárí“). Súčasná práca vo vysokom rozlíšení 2D a 3D postupne prináša možnosti k významnému zlepšeniu presnosti v rozpoznávaní tváre.



**Obrázok č. 5: Štandardná forma „eigenfaces“ a ich použitie [9]**

### *Geometria ruky/prsta*

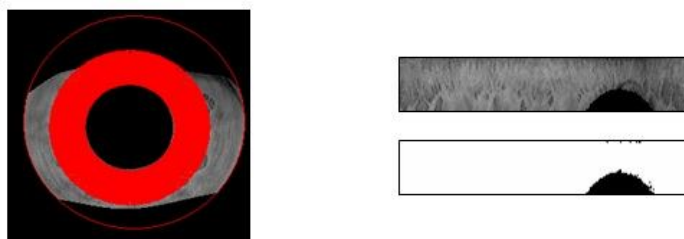
Jedným z prvých komerčne úspešných biometrických produktov bol systém pre rozpoznávanie geometrie ruky. Typický postup je nasledovný – užívateľ vloží PIN kód pre potvrdenie identity a potom vloží ruku na snímač, ktorý vytvorí obraz ruky (použitím zrkadiel obsahuje obraz pohľad zhora a zo strán). Následne sú odobraté rozmery a porovnané s tými, ktoré boli zozbierané pri registrácii.



**Obrázok č. 6: Systém pre snímanie geometrie ľudskej ruky**

### *Rozpoznávanie dúhovky ľudskeho oka*

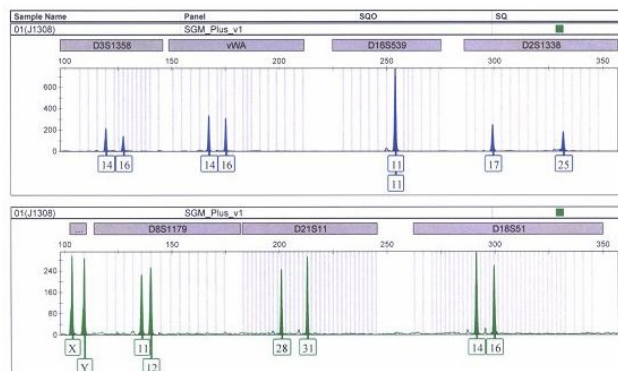
Táto metóda využíva techniku rozoznávania tvarov/vzorov na snímkoch dúhovky oka konkrétneho jedinca, vytvorením normalizovanej šablóny.



**Obrázok č. 7: Oblasť dúhovky „rozbalená“ do normalizovanej šablóny [10]**

Mnohé ďalšie identifikačné metódy sú v rozličných štádiách vývoja a/alebo komercializácie, označované ako doplnkové. Príkladom sú:

*Analýza DNA* – forma identifikácie často považovaná za primárnu biometrickú metódu, keďže DNA kód predstavujúci informáciu o identite je možné získať z každej bunky ľudskeho tela. Je založená na porovnaní alternatívnych foriem (prejavov génu) DNA postupností nájdených na identifikovateľných miestach v genetickom materiáli.



Obrázok č. 8: Profily DNA [11]

*Dynamika vlastnoručného podpisu* – meria rýchlosť a tlak vyvinutý pri vlastnoručnom podpise (nie ako podpis vyzerá).



a)



b)

Obrázok č. 9: (a) Rozpoznávanie dynamiky vlastnoručného podpisu, (b) Sledovanie dynamiky klávesových úderov [12]

*Dynamika klávesových úderov* – zaznamenáva šablónu písania konkrétneho jedinca.

*Rozpoznávanie držania tela* – sleduje ako niekto vyzerá pri chôdzi. Ako v prípade rozpoznávania tváre, aj táto technika je jednou z tých, ktoré človek intuitívne používa na rozpoznávanie blízkych osôb.

*Termografia ľudskej tváre* – meria ako sa teplo rozptyľuje (uniká preč) z tváre jedinca.

*Rozpoznávanie podľa tvaru ucha* – o túto metódu bol prejavovaný pomerne veľký záujem ako o jednu z metód rozpoznávania zo vzdialenosti.

*Pach (vôňa) tela* – nedávne pokroky v oblasti chemických analýz založených na polovodičoch viedli k vývoju „elektrických nosov“ schopných rozoznať asi 32 rozličných chemikálií.

*Rozpoznávanie sietnice* – vytvorí sa obraz očného pozadia a porovnáva sa s existujúcimi dátami uloženými v databáze.

### 2.2.3 Vzájomné porovnanie biometrických metód

V súčasnosti platí, že ľudské charakteristiky, či už fyzické alebo správania sa, môžu byť použité pre biometriu v prípade splnenia nasledujúcich parametrov:

- Všestrannosť/všeobecnosť (Universality) popisuje do akej miery je daná biometrická metóda unikátna vzhľadom k určitému súboru jedincov
- Unikátnosť (Uniqueness) udáva do akej miery sa daná biometrická metóda odlišuje svojimi charakteristickými črtami od ostatných
- Stálosť (Permanence) určuje mieru odolnosti použitej biometrickej metódy voči starnutiu
- Zhromažďiteľnosť (Collectability) predstavuje stupeň obtiažnosti získania vhodných použiteľných záznamov pri meraní
- „Výkonnosť“ (Performance) čiže presnosť, rýchlosť a robustnosť použitej technológie
- Prijateľnosť (Acceptability) miera osvedčenia technológie
- Odolnosť (Circumvention) stupeň obtiažnosti použitia náhrady, prípadne falšovania

Nasledující tabulka zobrazuje porovnání biometrických charakteristik vzhledem k daným parametrom:

**Tabulka č. 1: Porovnání biometrických charakteristik, podle A. K. Jain**

(H=High, M=Medium, L=Low)							
Biometrická charakteristika:	Všestrannost'	Unikátnost'	Stálost'	Zhromažditelnost'	„Výkonnost“	Prijatelnost'	Odolnost*
Odtlačky prstů	M	H	H	M	H	M	H
Tvář	H	L	M	H	L	H	L
Hlas	M	L	L	M	L	H	L
Geometria ruky	M	M	M	H	M	M	M
Důhová	H	H	H	M	H	L	H
DNA	H	H	H	L	H	L	L
Podpis	L	L	L	H	L	H	L
Klávesové údery	L	L	L	M	L	M	M
Chůdza	M	L	L	H	L	H	M
Termografie tváře	H	H	L	H	M	H	H
Tvar ucha	M	M	H	M	M	H	M
Vůň těla	H	H	H	L	L	M	L
Sítnice	H	H	M	L	H	L	H

\* - Odolnost' je označena právě opačnými barvami protože low je tímto případě žádúce namiesto high



### 3. TABLET

Táto kapitola je teoretickým úvodom do celej problematiky týkajúcej sa grafického tabletu, ako vstupného zariadenia pre získavanie dát (predstavujúceho samostatné zariadenie, nachádzajúce využitie len v symbióze s osobným počítačom, či akúsi integrovanú súčasť prenosného počítača označovaného ako Tablet PC), a postupne sú v nej priblížené všetky potrebné informácie - od historického pozadia až po dnešné aplikácie.

#### 3.1 ČO JE GRAFICKÝ TABLET ?

Zvyčajne je možné sa stretnúť s interpretáciou vedúcou k jasnému záveru a to, že sa jedná o nástroj pre digitalizáciu, či akýsi prostriedok pre kreslenie a čiastočné ovládanie osobného počítača. Hovoríme tak o alternatívnom type vstupného zariadenia, ktoré môže byť použité ako náhrada, prípadne ako doplnok k myši, guľčikovému ovládaču (označovanému ako „trackball“) a ostatným polohovacím zariadeniam. Samotný tablet pozostáva z dvoch základných častí – pracovnej plochy pre realizáciu kresieb a pera, stylusu alebo tzv. puku, určeného pre prácu s tabletom. V niektorých prípadoch sa môžeme dokonca stretnúť s bezdrôtovou myšou.



Obrázok č. 10: Grafický tablet Intuos3 (Wacom) [14]



### 3.2 ZARADENIE

Grafický tablet patrí do kategórie tzv. polohovacích zariadení, predstavujúcich akési vstupné rozhranie (špecificky „human interface device“ (HID) – teda zariadenia, ktoré typicky preberajú vstupné dáta od človeka, v snahe sprostredkovať adekvátny výstup) umožňujúce užívateľovi prenášať priestorové dáta do počítača. CAD systémy a grafické užívateľské rozhrania (GUI) poskytujú užívateľovi možnosť kontroly a následného predávania dát do počítača a to použitím fyzických pohybov a výrazov – ukázať, kliknúť, ťahať – realizovaných napr. pohybom myši (uchopenej v ruke) po povrchu fyzickej pracovnej plochy a aktivovaním príslušných tlačítok na tele myši. Pohyby polohovacieho zariadenia sú následne prenášané na obrazovku v podobe pohybu kurzora a ďalších vizuálnych zmien.

Zatiaľ čo najtypickejším zariadením tohto druhu je od nepamäti myš, do povedomia užívateľov sa čoraz častejšie dostávajú mnohé ďalšie. Aj napriek tomuto faktu práve myš zostáva akousi metaforou pre zariadenia, ktoré umožňujú pozorovateľný pohyb kurzora [15].

Pri väčšine polohovacích zariadení je pre predikciu rýchlosti, akou môže užívateľ dosiahnuť danú cieľovú pozíciu, základom Paul Fittsov zákon (publikovaný v roku 1954), dokonale popisujúci model ľudských pohybov. Je tak alfou a omegou samotného aktu ukazovania ako v reálnom svete (rukou či prstom), tak aj v prípade počítačov (myšou) [16].

Základné polohovacie zariadenia je možné, vychádzajúc z princípu činnosti, rozdeliť do troch skupín na zariadenia založené na:

#### 1. pohybe objektu

- *myš* - malé zariadenie, typicky držané v ruke a tlačené po horizontálnom povrchu predstavujúcom pracovnú plochu,



Obrázok č. 11: Myš Razer-Diamondback3G

- *mini-myš* - myš malých rozmerov (tvaru vajíčka) pre použitie s prenosnými počítačmi; zvyčajne sú jej rozmery dostatočne malé i pre použitie na povrchu samotného počítača; typicky sa jedná o optickú myš, vybavenú zatiahnuteľným pripojovacím káblom,



**Obrázok č. 12: Mini-myš Belkin (vľavo), Trackball Microsoft (vpravo)**

- „*trackball*“ - akýsi guľičkový ovládač, tvorený guľičkou s možnosťou rolovania v pevnom púzdre.

## 2. dotyku s povrchom

- „*touchpad*“ - hladký povrch schopný zaznamenať dotyk prstu; je normou pre moderné prenosné počítače; typicky je súčasťou prinajmenšom jedno tlačítko, no prosté kliknutie, na aké sme zvyknutí u myši, je možné previesť i poklepaním na samotný povrch,



**Obrázok č. 13: „Touchpad“ notebooku HP**

- *grafický tablet* – špeciálny tablet podobný „touchpadu“ avšak s tým rozdielom, že je ovládaný pomocou pera / stylusu, ktorý sa správa a používa ako bežné pero alebo ceruzka; palec ruky slúži zvyčajne pre kontrolu klikania a to cez dvojcestné tlačítko umiestnené na povrchu pera; problematika týkajúca sa grafického tabletu je rozpracovaná v nasledujúcich kapitolách,



**Obrázok č. 14: Grafický tablet Bamboo (Wacom) [14] (vľavo),  
dotyková obrazovka (vpravo)**

- *dotyková obrazovka „touchscreen“* – typicky umiestnená na monitore, podobá sa akémusi štítu; toto zariadenie využíva softvérovú kalibráciu pre zabezpečenie zhody medzi polohou na obrazovke a polohou kurzoru; mnohé firmy sa snažia začleniť tento prvok do už existujúcich displejov a zariadení akými sú napr. prenosné počítače.

### 3. relatívnej polohy objektu

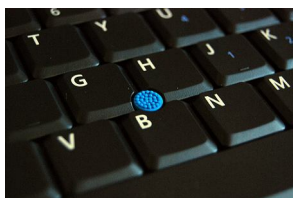
- *Izotonický pákový ovládač* – kde môže užívateľ voľne meniť polohu páky, pôsobením viacmenej konštantnej sily; do tejto skupiny patrí pákový ovládač („joystick“) a analógový ovládač,



**Obrázok č. 15: Pákový ovládač Cyber Knight (Rockfire) (vľavo),  
analógový ovládač (vpravo)**

- *Izometrický pákový ovládač* – kde užívateľ ovláda páku zmenou veľkosti sily, ktorou na ňu pôsobí, a poloha páky zostáva viacmenej nezmenená; patrí

sem polohovacia páka – akýsi výstupok citlivý na tlak, používaný u prenosných počítačov; typicky umiestnený medzi klávesami g, h a b [15].

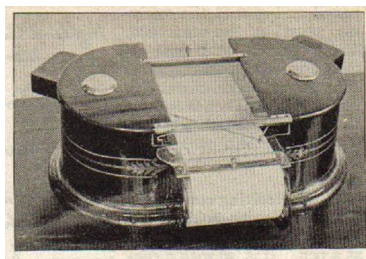


**Obrázok č. 16: Umiestnenie uvedeného typu ovládača**

### 3.3 HISTORICKÝ VÝVOJ

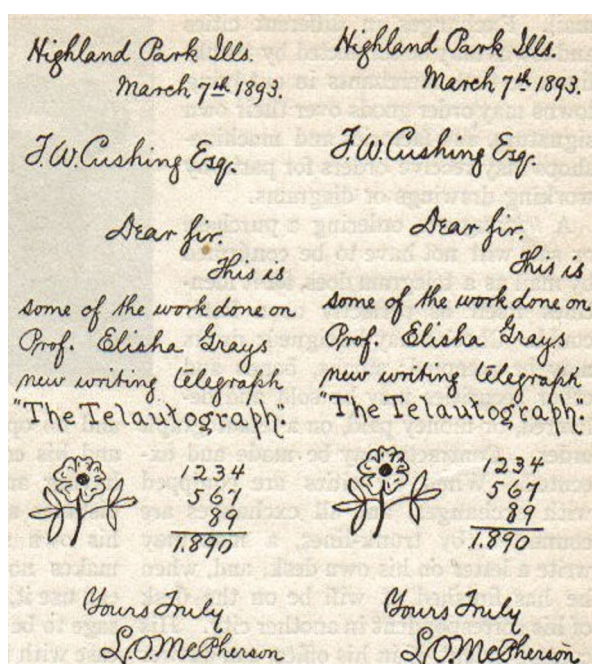
Pri pohľade do histórie, a s cieľom dopracovať sa k počiatkom grafického tabletu ako takého, dospejeme v prvom rade k zisteniu, že prvé grafické tablety boli zväčša známe pod pojmom akustické tablety. Toto označenie je úzko späté so samotným princípom - použitie zvuk emitujúceho pera a mikrofónov pre jeho záznam. Aktuálna pozícia stylusu môže byť následne určená na základe rozdielu časov zaznamenaného zvuku. Zvukom môže byť napr. malý elektrický výboj, ktorý vznikne pri umiestnení stylusu do blízkosti pracovnej plochy tabletu, generovaný prvkom podobným štartovacej sviečke používanej v automobiloch. Tento systém bol pomerne zložitý, nákladný a náchylný na šum v prostredí [22].

Prvým elektronickým tabletom, ako ich poznáme dnes, bol tzv. teleautograf (Obrázok č. 17), patentovaný v roku 1888; autorom bol Elisha Gray, známy popri Alexandrovi Grahamovi Bellovi ako súdobý vynálezca telefónu.



**Obrázok č. 17: Model teleautografu z roku 1893 [17]**

Profesor Gray si ako jeden z mála uvedomil, a zároveň sprostredkoval verejnosti, potrebu tvorby lepšej a odlišnej triedy služieb; potrebu vzniku niečoho, čo by nahradilo ako telegraf tak aj telefón a umožnilo prenášať písané slovo rýchlosťou telegramu. Výsledkom jeho úsilia bol prístroj označený ako teleautograf - nástroj na prenos ručne písaného textu či kresieb pomocou elektriny. Reprodukcia bola v prijímacej stanici zabezpečená prostredníctvom špeciálneho pera, ktorého pohyby boli ovládané ceruzkou určenou pre tvorbu textu či kresieb v stanici pre odosielanie [17].



Obrázok č. 18: Pôvodný a prenesený text (teleautograf) [17]

Prvý grafický tablet, podobný súčasným zariadeniam, a použitý pre rozpoznávanie ručne písaného textu bol tzv. Styalator z roku 1957. Známejší, a často nesprávne označovaný ako prvý digitizér, je RAND tablet alebo tiež Grafacon (Obrázok č. 19 na nasledujúcej strane), predstavený v roku 1964. Daný tablet využíval mriežku tvorenú drátmi, umiestnenú pod povrchom podložky (pracovnou plochou), za účelom získania horizontálnych a vertikálnych súradníc pri pôsobení slabého magnetického signálu. Stylus prijímal magnetický signál a následne dochádzalo k jeho dekódovaniu a získaniu informácie o polohe, teda súradníc.





**Obrázok č. 19: RAND tablet (Grafacon) [18]**

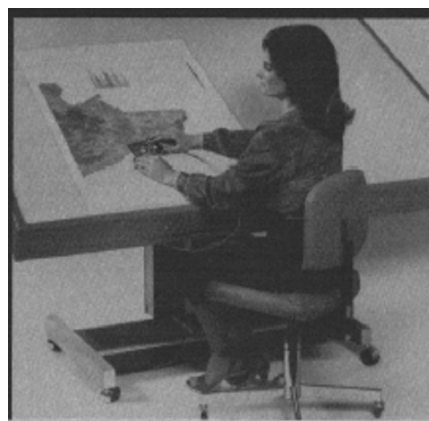
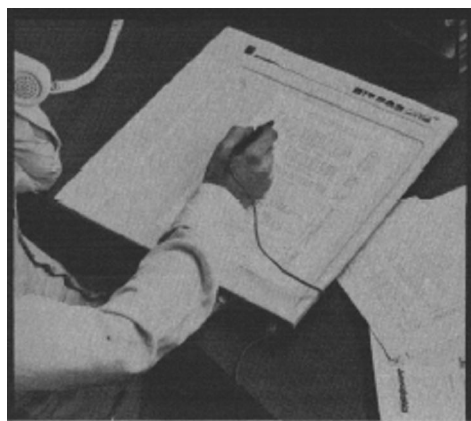
Grafické tablety, pre bežné použitie v domácnostiach, sa začali objavovať čoraz častejšie po príchode tabletu označeného ako KoalaPad. Ten bol od roku 1984 produkovaný a propagovaný firmou Koala Technologies pre niekoľko prvých 8-bitových domácich počítačov ako Apple II, TRS-80 Color Computer, Atari 8-bit, Commodore 64 a tiež IBM PC. Pôvodne bol daný model navrhnutý Dr. Davidom Thornburgom ako finančne nenáročná učebná pomôcka pre školy. Dosahoval pritom rozmery 4" x 4", obsahoval 2 tlačítka umiestnené nad vlastnou plochou tabletu a pre pripojenie k počítaču bol použitý port pre pákové ovládače, čím bolo dosahované pomerne nízke rozlíšenie. Namiesto stylusu bolo možné, za účelom kreslenie či prácu v ponuke, použiť prsty. Postupom času sa objavili mnohé modifikácie tohto prevedenia, ako napr. tablet Atari CX77 Touch Tablet, v prípade ktorého bola adresácia tlačítok umiestnených na tablete realizovaná pomocou stlačiek vstavaných priamo na styluse. Takáto úprava sa medzi užívateľmi stala veľmi populárnou a položila základy nového trendu dizajnu stylusov [19].



**Obrázok č. 20: Prospekty a samotný KoalaPad**

Od polovice 80. rokov sa k základným funkciám tabletu, teda tvorbe textu a kresieb, začali pridávať i funkcie doplnkové, napr. rozpoznávanie ručne písaného textu či mnohé rozširujúce ponuky umiestnené priamo na pracovnej ploche tabletu.

Tablety vo funkcii digitizérov sa dostali do povedomia v polovici 70. a začiatkom 80. rokov 20. storočia, ako dôsledok komerčného úspechu inteligentného digitizéra ID a Bit-Padu od firmy Summagraphics Corp. Tieto produkty boli použité ako vstupné zariadenia pre mnohé CAD systémy a rovnako aj pre softvéry akým je napr. AutoCAD. Jednalo sa o tablety založené na magnetostrieknej technológii, využívajúcej dráty vyrobené zo špeciálnej zliatiny, natiahnuté cez tuhý podklad a slúžiace pre presnú lokalizáciu špičky stylusu alebo stredu kurzoru na pracovnej ploche. Táto technológia umožňovala taktiež prácu s osou "Z", teda priblížením. Na Obrázku č. 21 (vpravo) je Bit-Pad One, o rozmeroch 11" x 11" a rozlíšení 0.004"; séria ID ponúkala podobné rozlíšenie, avšak pri rozmeroch až 42" x 60"; produkt s označením Summagrid prišiel s rozlíšením 0.001" na podsvietených tabuliach s rozmermi 42" x 60" (Obrázok č. 21 vľavo) [20].



**Obrázok č. 21: Summagrid na dobovej fotografii [20] (vľavo), Bit-Pad One na dobovej fotografii [20] (vpravo)**

### 3.4 SÚČASNOSŤ

Väčšina moderných grafických tabletov funguje na podobnom princípe ako RAND tablety (kapitola 3.3). V zariadeniach dnes bežne používaných tak možno nájsť len malé odlišnosti, pričom sa jedná o vylepšenia a zdokonalenia už používaných výrobných postupov.

V prvom rade je nutné spomenúť úpravu drátovej mriežky, kedy došlo k oddeleniu horizontálnych a vertikálnych vlákien tenkou vrstvou izolantu. Pri pôsobení tlaku na pracovnú plochu grafického tabletu dochádza k stretu horizontálneho a vertikálneho vlákna, teda označeniu daného bodu mriežky, a oboma vláknami preteká elektrický prúd. Keďže prúd preteká vždy len dvoma vláknami, ktoré sú aktuálne v kontakte, je možné získať súradnice polohy stylusu. Správnemu určeniu napomáha taktiež slabé magnetické pole vytvárané vo vzdialenosti 1" od pracovnej plochy [21].

#### 3.4.1 Parametre

##### Rozmery

Práve veľkosť je jedným z prvých faktorov, ktoré je v prípade hľadania vhodného grafického tabletu, nutné zvážiť. Platí totiž pravidlo – väčší neznamená lepší – pričom záleží v prvom rade na cieľovej skupine. Pre bežných užívateľov sú typické rozmery 4" x 5" a 6" x 8". Užívatelia pracujúci s CADom, umelci a technickí pracovníci zvyčajne vykazujú potrebu väčšieho povrchu pre realizáciu ich tvorivej činnosti, avšak so zväčšujúcim sa povrchom narastá i cena. Veľa ľudí uprednostňuje menšie rozmery, keďže nevyžadujú žiadne extrémne pohyby rúk. Na druhej strane, práve táto skutočnosť môže prekážať umelcom, zvyknutým pri svojej činnosti na voľnosť pohybu po pracovnej ploche. Dôležitou skutočnosťou, ktorú je nutné brať do úvahy, je informácia o rozmeroch tabletu. Tá vždy udáva rozmery fyzickej pracovnej plochy, pričom rozmery plochy prenesenej pri spracovaní môžu byť podstatne väčšie. Ako príklad možno uviesť tablet Wacom Intuos s rozmermi pracovnej plochy 6" x 8" a prenášanou plochou o rozmeroch 10" x 13.5".



V poslednej dobe sú populárne tablety s rozmermi 4" x 5", 6" x 8" a 9" x 12", ktoré takmer odpovedajú pomeru strán 4:3 bežne používaných monitorov.

Po roku 2000 sa začali šíriť a používať širokohlé monitory. V dôsledku toho začali firmy ako Wacom a Aiptek produkovať tablety so širokohlým formátom, v snahe dosiahnuť lepšiu podobnosť s pomerom strán širokohlých monitorov a tiež v prípade použitia viacerých monitorov. Pre správne zobrazovanie na monitore sa potom stará príslušný softvér.

### **Rozhranie**

Rozhraním sa rozumie, ako je tablet pripojený k osobnému počítaču. Väčšina tabletov je vybavená rozhraním USB, čo umožňuje jednoduchú pripojiteľnosť a manipuláciu. Pre kompatibilitu so staršími počítačmi je možné nájsť i varianty so sériovým portom.

Ďalšou možnosťou pripojenia je bezdrátové pripojenie pomocou technológie Bluetooth.

### **Pero/Stylus a príslušenstvo**

Pero/Stylus je navrhované v snahe dosiahnuť komfort a prirodzenosť pri práci. Prípadná potreba napájania vedie k použitiu batérií, teda k zvýšeniu hmotnosti a potreby údržby, no typicky napájanie nie je potrebné a pero je možné nahradiť takmer ľubovoľným objektom. Samotné pero je však vždy navrhované tak, aby vyhovovalo danému zariadeniu a bolo možné pri jeho správnom používaní dosiahnuť maximálnu životnosť zariadenia, preto treba brať tento fakt do úvahy. Na povrchu pera je obvykle možné nájsť prepínač alebo tlačítka, a mnohé perá sú vybavené časťou pre mazanie. Jednotlivé tlačítka sú pritom programovateľné - špecifické funkcie ako pravý klik či dvojklik.

### **Citlivosť na tlak a ďalšie charakteristiky**

Úroveň tlaku odpovedá citlivosti na tlak na samotnom povrchu pracovnej plochy tabletu. Väčšinou sa stretávame s 256, 512 alebo 1024 úrovňami tlaku.

Citlivosť na tlak môže ovládať hrúbku čiar, priehľadnosť alebo farbu. Čím vyššia je daná citlivosť, tým hodnovernejšie reaguje tablet na vonkajšie podnety a poskytuje tak rozsiahle možnosti ovládania [22].

Ďalšími charakteristikami, ktorá je možné pri použití pera sledovať, sú napr. natočenie samotného pera voči pracovnej ploche (typicky v rozsahu  $\pm 60^\circ$ ), či v prípade špeciálneho, tzv. „umeleckého pera“, dokonca jeho rotácia okolo zvislej osi ( $0^\circ$  až  $360^\circ$ ). V oboch prípadoch je nutné zdôrazniť, že daná charakteristika musí byť použitým grafickým tabletom podporovaná.

### 3.4.2 Rozdelenie

Grafické tablety je možné kategorizovať na základe viacerých kritérií. Dve z možností sú uvedené nižšie.

Delenie podľa princípu využívaného pre činnosť na:

**Optické tablety** - využívajú pre svoju činnosť malú digitálnu kameru umiestnenú na pere, a následné porovnávanie vzorov v obraze. Najúspešnejším príkladom je technológia vytvorená spoločnosťou Anoto.

**Akustické tablety** - jedny z prvých modelov boli označované ako „iskrové“ tablety, malý generátor zvuku bol umiestnený na pere a zvuk zachytávaný mikrofónmi umiestnenými blízko pracovnej plochy. Moderné varianty sú schopné činnosti v troch dimenziách. Aj napriek tomu, že tento druh tabletov je častokrát považovaný za zastaralý a nepresný (v porovnaní s dnešnými technológiami), neustále nachádza využitie v určitých špecializovaných produktoch, napr. od firmy Pegasus, či firmy PNF – digitálne pero „DUO for laptop“ [22], umožňujúce premeniť ľubovoľnú plochu (displej notebooku, klasický papier apod.) na akúsi pomyselnú pracovnú plochu grafického tabletu. A práve táto skutočnosť je zároveň hlavným dôvodom stále pretrvávajúcej obľúbenosti medzi užívateľmi.

**Elektromagnetické tablety** - na tomto princípe generovania a detekcie elektromagnetického signálu pracuje väčšina dnes známych zariadení. Typickým zástupcom sú tablety firmy Wacom, Pencept apod.

**Kapacitné tablety** - boli navrhnuté pre využitie elektrostatických alebo kapacitných signálov, pričom typickým predstaviteľom danej technológie je spoločnosť Scriptel.

Na rozdiel od kapacitného princípu, používaného napr. v prípade dotykových obrazoviek, produkty danej spoločnosti vychádzajú z princípu detekcie elektrostatického signálu, čo vedie k schopnosti detekcie polohy pera pri jeho priblížení, či prostom vznášaní sa nad pracovnou plochou.

V prípade všetkých uvedených technológií by malo byť možné získať i informáciu o vzdialenosti pera od pracovnej plochy, prípadne uhle natočenia apod.

Delenie podľa potreby napájania na:

**Pasívne tablety** - najvýraznejšie produkty firmy Wacom, vychádzajú z použitia elektromagnetickej indukcie. Horizontálne i vertikálne vlákna siete pracujú ako prijímacie i vysielačie cievky (rozdiel oproti sieti použitej pri RAND tabletoch, kedy mali vlákna len schopnosť vysielať). Tablet sám o sebe generuje elektromagnetický signál, ktorý je zachytávaný LC obvodom v používanom pere. Vlákna siete následne zmenia svoj stav, dochádza k prepnutiu do stavu prijímania a čítaniu signálu vytvoreného perom. Moderné prevedenia umožňujú taktiež sledovať úroveň pôsobiaceho tlaku a využívať jedno alebo viac tlačítok umiestnených na tele pera (pre tento účel slúži elektronika umiestnená priamo v pere). V prípade starších tabletov dochádzalo zmenou tlaku na pero, prípadne stlačením prepínača, k zmene vlastností LC obvodu a ovplyvňovaniu vlastností signálu. U moderných prevedení je zvyčajne digitálny tok dát prevádzaný na signál.

Použitím elektromagnetického signálu je tablet schopný zaznamenať polohu pera a to i bez potreby kontaktu s pracovnou plochou a zároveň i bez potreby dodatočného napájania batériami. Patent firmy Wacom však neumožňuje využitie tejto techniky inou spoločnosťou.

**Aktívne tablety** - odlišnosť oproti predchádzajúcej kategórii spočíva v prevedení pera, ktoré obsahuje elektroniku generujúcu a vysielajúcu signál (samozrejme s potrebou napájania). Tieto perá sú odkázané na napájanie batériou a teda i robustnejšie. Eliminácia potreby napájania znamená zabezpečiť použitie len prijímacieho módu tabletu, čím odpadá možný vznik čiastočného chvenia spôsobeného prepínaním medzi stavom vysielania a prijímania [22].

### 3.4.3 Použitie

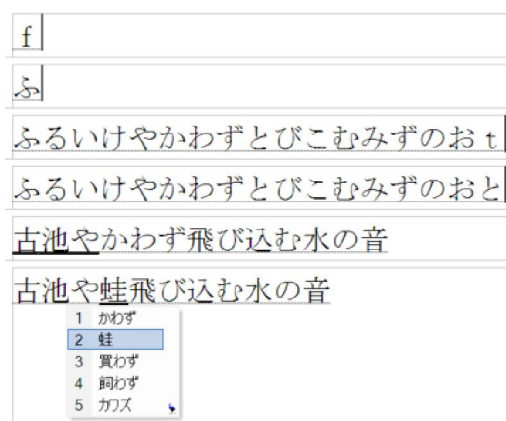
Grafické tablety, vďaka svojej konštrukcii a funkčnosti umožňujúcej prácu s perom a detekciu úrovni tlaku či iných interakcií s pracovnou plochou, sú častokrát považované za prirodzenú cestu tvorby počítačovej grafiky, predovšetkým dvojrozmernej. Použitie pre daný účel má svoje korene už v dobe vzniku prvých grafických tabletov, príkladom je softvér vytvorený pre RAND tablet (kapitola 3.3). Nastolený trend, v snahe uľahčiť prácu cieľovej skupine užívateľov, pokračoval aj v nasledujúcich rokoch a výsledkom bol grafický editor s označením KoalaPainter navrhnutý pre prvý grafický tablet pre bežné použitie KoalaPad z roku 1984.



Obrázok č. 22: Ponuka grafického editoru KoalaPainter [19]

Unikátnou črtou bola v tomto prípade možnosť uložiť do pamäte počítača súčasne dva obrázky, s možnosťou ich vzájomného prepínania, umožňujúc sledovať rozdiely medzi pôvodným a upraveným obrázkom, či využívať prvky ako kopírovať a prilepiť pomocou dvoch tlačítok na tele tabletu. Následne, v roku 1988, sa začala písať história dnes dobre známeho grafického editoru Photoshop spoločnosti Adobe, a v roku 1989 nemenej známeho editoru CorelDRAW spoločnosti Corel Corporation. V roku 1995 sa dokonca objavil prvý voľne šíriteľný editor pod GPL, tzv. GIMP, vytvorený skupinou nadšencov. Posledné uvedené editory sú v súčasnosti prispôsobené a často využívané pre prácu s grafickými tabletmi, umožňujúc pritom využitie úrovni tlaku (a v niektorých prípadoch dokonca naklonenia pera či jeho natočenia) pre účely grafickej tvorby. Uvedené črty sú pritom použité pre zmenu veľkosti kresliaceho nástroja, jeho tvaru, farby apod.

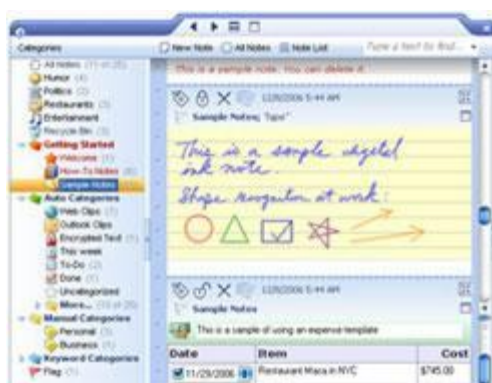
Vo východnej Ázii sú grafické tablety používané v spojení s editačným softvérom (tzv. „input method editor“ (IME) pre vkladanie znakov, ktoré nie sú dostupné na bežných vstupných zariadeniach) pre písanie čínskych, japonských a kórejských znakov (tzv. CJK). Táto technológia je veľmi obľúbená, pomerne lacná a ponúka možnosť pracovať s počítačom prirodzenejším spôsobom ako použitím klávesnice a myši [22].



Obrázok č. 23: Grafické rozhranie IME

Podobná aplikácia grafických tabletov, s cieľom zjednodušiť a zefektívniť prácu s PC pri bežnom použití, je pri rozpoznávaní ručne písaného textu. Jedná sa pritom o postup získania a interpretovania zrozumiteľného ručne písaného textu zo vstupného zariadenia akým je napr. tablet. Hovoríme tak o on-line prístupe, kedy dochádza k automatickej konverzii písaného textu. Tento druh dát je známy pod pojmom digitálny atrament a možno ho považovať za akúsi dynamickú reprezentáciu ručne vytvoreného textu.

Získaný signál je následne prevedený na popisné kódy, použiteľné pre počítač a aplikácie určené pre spracovanie textu. Príkladom aplikácie, použiteľnej pre dané účely, je napr. Microsoft OneNote, či jeho konkurent EverNote.



Obrázok č. 24: Grafické rozhranie programu EverNote

Spôľahlivá autorizácia (authorization) a taktiež overovanie (authentication) sa v súčasnosti stávajú nevyhnutnými pre mnohé každodenné činnosti akou je napr. prevádzanie finančných operácií a pod. Rozpoznávanie jedincov je zásadnou aktivitou vo vnútri našej spoločnosti a kultúry, odkedy je pre mnohé aplikácie zaručenie identity a autentičnosti zainteresovaných ľudí nevyhnutnosťou. Tento prístup si preto vyžiadala vznik systémov pre overovanie tvorby vlastnoručného podpisu (on-line metóda) alebo už výsledného produktu (off-line metóda).

*On-line:* je sledovaný proces vzniku podpisu, tzn. dynamika - rýchlosť, poradie jednotlivých čŕt, ich počet a tlak, pričom sú získané unikátne biometrické dáta a následne po ich spracovaní i jedinečný vzor charakterizujúci danú osobu - pre získanie potrebných čŕt je nevyhnutné buď použitie grafického tabletu, prispôsobeného pre tento účel (príkladom je napr. produkt SignDoc od spoločnosti Softpro), prípadne použitie špeciálneho pera pre podpis na ľubovoľnom povrchu, tzv. Bio-Pen od spoločnosti Dynamic Biometric Systems, Inc., vybaveného rozhraním USB; samozrejmosťou je potreba použitia špecifického softvéru pre spracovanie dát. (Obrázky č. 29 a 30 na nasledujúcej strane)



**Obrázok č. 25: SignDoc spoločnosti Softpro (vľavo), Bio-Pen spoločnosti DBS, Inc. (vpravo)**

*Off-line:* je spracovávaný už výsledný produkt, pracuje sa teda s hotovým obrazom podpisu, a dochádza k analýze geometrických vzorov; pre získanie dát postačí podpis vykonaný klasickým perom na papier, no vylúčené nie je ani použitie grafického tabletu schopného podpis zaznamenať

Grafické tablety sú taktiež veľmi populárne v spojení s technickým kreslením a CAD aplikáciami, keďže je možné na ne položiť hárok papiera bez obmedzenia funkčnosti.

Výnimkou nie je ani umenie. Mnohí úspešní tvorcovia webkomiksov, ako napr. Hawk (komiks AppleGeeks) a ďalší, sa rozhodli pre túto cestu z dôvodu jednoduchosti návrhu a úpravy, v porovnaní s dovtedajšou tvorbou na papier.

Naviac, stále viac umelcov častokrát začínajúcich pri klasických metódach tvorby, začína využívať výhody tohto prístupu [22].





**Obrázok č. 26: Dielo mladej rumunskej umelkyne (použitie grafického tabletu)  
(vľavo), Komiks AppleGeeks (vpravo)**

Dokonca i osoby nevyužívajúce tablet za účelom tvorby umenia sa častokrát rozhodnú pre túto variantu vstupného zariadenia. Veľkým lákadlom je práve intuitívnejšie ovládanie, keďže pozícia pera na pracovnej ploche tabletu typicky odpovedá umiestneniu kurzora v danom grafickom rozhraní, zobrazenom na obrazovke.

#### 3.4.4 Predchádzanie vzniku ochorení

Zástancovia tabletov propagujú túto variantu, samozrejme ak to dovoľuje dané použitie, ako náhradu ostatných typov polohovacích zariadení. Dôvod je jednoduchý – ergonomickejšia metóda ovládania, vedúca k zníženiu pravdepodobnosti vzniku ochorenia (označovaného ako RSI „repetitive strain injury“ a pokrývajúceho celý súbor poškodení, ktoré sú vyvolané prácou v neergonomickom prostredí) a vedúceho k napr. poškodeniu hybnosti ramenných pletencov, poškodeniu šliach prstov a pod. Jedná sa pritom o problém súčasnosti, kedy je RSI považované za jednu z najčastejších príčin vzniku choroby z povolania v kancelárskych profesiách.



**Obrázok č. 27: Pohľad na usporiadanie klasickej kancelárie**



## 4. ZÍSKAVANIE DÁT Z GRAFICKÉHO TABLETU

Daná kapitola je venovaná popisu možnosti získavania dát, a to ako z grafického tabletu v kombinácii s bežným osobným počítačom tak aj z Tablet PC, v prostredí operačného systému Windows.

### 4.1 TABLET PC

Je možné konštatovať, že sa jedná o notebook, prenosný počítač, vybavený dotykovou obrazovkou alebo integrovaným grafickým tabletom, slúžiacim na kompletne ovládanie počítača prostredníctvom stylusu alebo prstov, vystupujúcich v úlohe klávesnice a myši. Táto skutočnosť poskytuje užívateľom nebyvalú voľnosť a flexibilitu pri práci. Zariadenia tohto druhu sú tak využívané všade tam, kde bežné notebooky nedosahujú požadovanú mieru praktickosti, či dostatočnú funkčnosť.

#### 4.1.1 SDK

Komponenty platformy Tablet PC umožňujú vyvíjať aplikácie navrhnuté pre kooperáciu so samotným hardvérom Tablet PC, či grafického tabletu pripojeného k počítaču.

Môžu tak byť použité pre tvorbu aplikácií v prostredí Windows XP Tablet PC Edition 2005, Microsoft Windows Vista, Microsoft Windows 7 a niektorých ďalších operačných systémov.

Softvérový vývojový balík (Software Development Kit, SDK) pre Tablet PC je v súčasnosti dostupný vo verzii 1.7, ktorá priniesla v porovnaní s poslednou verziou, niekoľko rozšírení. Jedná sa predovšetkým o pridanie podpory rozhrania pre programovanie nesúceho názov StylusInput, poskytujúceho prístup k toku dát, a rovnako podpory pre využitie týchto prvkov priamo na internetových stránkach. Navyše došlo k redukcii počtu potrebných assembly<sup>1</sup> a vytvoreniu jednej, nesúcej označenie Microsoft.Ink.

---

<sup>1</sup> Globálne označenie pre súbor automaticky generovaný po úspešne prevedenej kompilácii každej aplikácie v .NET. Môže sa tak jednať o dynamicky spojenú knižnicu (Dynamic Link Library, DLL), ako je tomu v tomto prípade, alebo spustiteľný súbor (Executable file, EXE).

Od verzie operačného systému Microsoft Windows Vista je Tablet PC SDK navyše priamo súčasťou SDK daného systému, voľne dostupného na oficiálnych stránkach spoločnosti Microsoft, v aktuálnej verzii 7.0 určenej primárne pre Windows 7.

## 4.2 POUŽITÝ GRAFICKÝ TABLET

Cieľom podkapitoly je priblížiť grafický tablet Trust TB-7300, samozrejme čo do vlastností kľúčových pre riešenie danej problematiky.



**Obrázok č. 28:** Použitý grafický tablet

Kľúčové vlastnosti sú nasledujúce:

- Jedná sa o širokouhlý grafický tablet s rozmermi 30,5 x 19,5 cm.
- Súčasťou balenia je ergonomické dvojtláčkové pero, poskytujúce možnosť rozlíšiť 1024 úrovní tlaku na pracovnú plochu tabletu.
- Je navrhnutý ako pre užívateľov pravákov, tak i pre ľavákov.
- Pracovná plocha je chránená ochrannou plastovou fóliou, pod ktorú je navyše možné vložiť napr. podklad so vzormi určenými pre obresľovanie a zabrániť tak jeho náhodným posuvom v priebehu sedenia, či pri zmene testovaných subjektov.

### 4.3 .NET FRAMEWORK

Je softvérový framework<sup>2</sup>, použitelný v prostředí operačního systému Microsoft Windows. Obsahuje rozsiahlu knižnicu hotových riešení bežných programátorských problémov a virtuálny stroj (virtualizované prostredie medzi platformou počítača a operačným systémom), ktorý spravuje spúšťanie programov vytvorených špeciálne pre daný framework. .NET Framework je dielom spoločnosti Microsoft a je určený pre použitie vo väčšine nových aplikácií pre platformu Windows.

Základná knižnica tried Base Class Library (BCL) poskytuje širokú škálu možností, zahŕňajúc pritom užívateľské rozhranie, prístup k dátam, spojenie s databázami, kryptografiu, vývoj internetových aplikácií, numerické algoritmy a sieťovú komunikáciu. Je preto programátormi používaná za účelom tvorby aplikácií, samozrejme za súčasného kombinovania s ich vlastným kódom.

Program napísaný pre .NET Framework sa spúšťa v softvérovom prostredí, spravujúcom požiadavky pri samotnom spustení a následne počas doby jeho behu. Ako súčasť frameworku je toto prostredie označované Common Language Runtime (CLR). Vytvára zdanie spustenia virtuálneho stroja pre danú aplikáciu, vďaka čomu programátor nie je nútený brať ohľad na možnosti dostupného procesoru. Okrem toho zodpovedá CLR za bezpečnosť, správu pamäti a obsluhu výnimiek. Spolu s BCL tak tvorí podstatu celého frameworku.

Počiatky vývoja samotného frameworku siahajú do obdobia konca 90. rokov minulého storočia, pôvodne sa jednalo o projekt nesúci názov Next Generation Windows Services (NGWS), a prvá beta verzia s označením .NET 1.0 bola vydaná koncom roku 2000. Informačný prehľad ďalšieho vývoja je uvedený v Tabuľke č. 2 (na nasledujúcej strane).

---

<sup>2</sup> Forma abstrakcie, v ktorej daný kód poskytujúci obecnú funkčnosť, môže byť jednoducho nahradený / upravený užívateľským kódom poskytujúcim určitú špecifickú funkčnosť. Jedná sa tak o špeciálny prípad softvérových knižníc.

**Tabuľka č. 2: Prehľad dostupných verzií .NET Framework**

Verzia	Číslo verzie	Dátum vydania	Visual Studio	Štandardne vo Windows
<b>1.0</b>	1.0.3705.0	2002-02-13	VS .NET	--
<b>1.1</b>	1.1.4322.573	2003-04-24	VS .NET 2003	Windows Server 2003
<b>2.0</b>	2.0.50727.42	2005-11-07	VS .NET 2005	--
<b>3.0</b>	3.0.4506.30	2006-11-06	--	Windows Vista, Windows Server 2008
<b>3.5</b>	3.5.21022.8	2007-11-19	VS .NET 2008	Windows 7, Windows Server 2008 R2
<b>4.0</b>	4.0.30319.1	2010-04-12	VS .NET 2010	--

S postupným vývojom sa navyše objavovali komponenty, ktoré mali prispieť k zefektívneniu práce programátora a zvýšeniu príťažlivosti pre koncového užívateľa.



**Obrázok č. 29: Prehľad verzií frameworku, spolu s komponentami [24]**

Ako vidieť na Obrázku č. 29, už framework **2.0** priniesol rozšírenia v podobe:

- **WinForms** - označení rozhrania pre programovanie (Appliaction Programming Interface, API<sup>3</sup>), umožňujúceho tvorbu grafického užívateľského rozhrania,
- **ASP.NET** - nástroje pre vývoj internetových aplikácií,
- **ADO.NET** - prístup k dátam a dátovým službám.

Verzia **3.0** pozostáva zo štyroch nových komponentov:

- **Windows Presentation Foundation (WPF)** - rozhranie podobné WinForms, no založené na XML a vektorovej grafike,
- **Windows Communication Foundation (WCF)** - systém pre tvorbu komunikačnej infraštruktúry aplikácií,
- **Windows Workflow Foundation (WF)** - nástroj pre definovanie sekvenčných procesov pri použití schém („workflows“),
- **Windows CardSpace** - uchovávanie informácií o identite jedinca, správa operácií ako prihlásenie sa do systému apod.

Verzia **3.5** je typická dvoma komponentami:

- **Language Integrated Query (LINQ)** - obecný nástroj pre manipuláciu s rôznymi dátami – objekty, databázy a XML súbory,
- **ADO.NET** – rozšírenie možností pre daný komponent.

Verzia **4.0** vo svojej finálnej podobe prináša:

- **Parallel Extension** - paralelných výpočtov, so zameraním na viacjadrové a distribuované systémy
- **nové prvky** pre jazyky VB.NET a C#
- **plnú podporu** pre IronPython, IronRuby a F# [24, 25].

---

<sup>3</sup> Rozhranie implementované daným softvérom s cieľom zabezpečiť komunikáciu s ostatnými programami, podobne ako je tomu v prípade užívateľského rozhrania pre interakciu medzi užívateľom a počítačom.

Aj napriek faktu, že .NET Framework predstavuje dominantnú formu implementácie .NET technológií, existujú i niektoré ďalšie prevedenia, napr.: [24]

- **Microsoft .NET Micro Framework** – je platforma určená pre zariadenia s obmedzeným výpočtným výkonom
- **Microsoft .NET Compact Framework** – je platforma určená pre mobilné zariadenia používajúce operačný systém Windows CE, teda mobilné telefóny, PDA apod.
- **Mono** – je implementáciou CLI a častí .NET BCL, ktorá má duálnu licenciu ako slobodný a zároveň i ako proprietárny softvér, pre operačné systémy unixového typu

#### 4.3.1 WinForms

Je skratkou pre spojenie Windows Forms, používané ako označenie rozhrania pre programovanie, umožňujúceho tvorbu grafického užívateľského rozhrania za využitia subsystému pre reprezentáciu grafických objektov do výstupných zariadení, tzv. Graphics Device Interface (GDI). Súčasťou .NET Frameworku je od verzie 2.0 (Obrázok č. 30 na nasledujúcej strane).

Poskytuje prístup k základným prvkom operačného systému Microsoft Windows a to prostredníctvom „obaľovania“ existujúcich programovacích rozhraní (API) do spravovaného kódu (managed code<sup>4</sup>).

Výsledná Windows Forms aplikácia je aplikácia riadená udalosťami, teda väčšinu času sa jednoducho čaká na užívateľa a nejakú jeho činnosť, vyplnenie textového okna, kliknutie na tlačítko apod. [26].

---

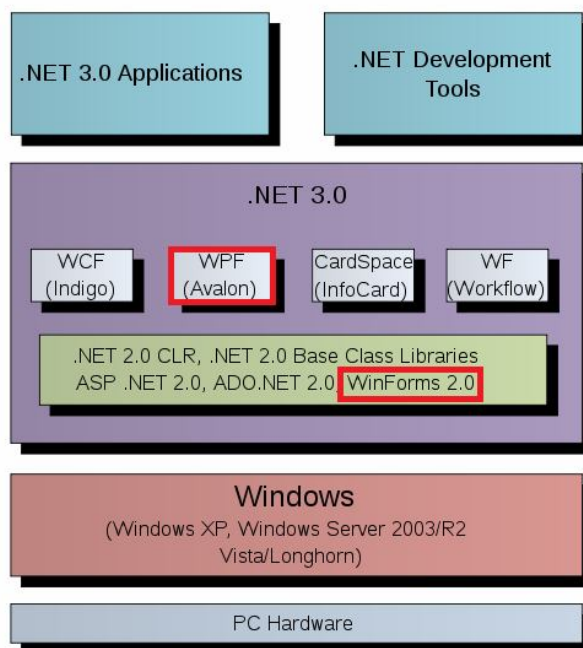
<sup>4</sup> Označenie zavedené spoločnosťou Microsoft pre odlišenie zdrojového kódu, vyžadujúceho pre spustenie správu zo strany virtuálneho stroja vytvoreného CLR (kapitola 4, časť 4.3).

#### 4.3.2 Windows Presentation Foundation (WPF)

Je grafický subsystém pre renderovanie užívateľského rozhrania v aplikáciách určených primárne pre operačný systém Windows. WPF bol pôvodne vydaný ako časť .NET Framework verzie 3.0, za účelom odstránenia závislosti na starom subsystéme GDI. Je postavený na DirectX, poskytujúcom hardvérovú akceleráciu a umožňujúcom použitie moderných črt užívateľského rozhrania ako priehľadnosť apod. Poskytuje tak programovací model pre tvorbu aplikácií, pri súčasnom zabezpečení oddelenia užívateľského rozhrania a logiky pre správu a výmenu informácií.

WPF taktiež ponúka nový jazyk, známy pod označením XAML, ktorý je alternatívou pre definovanie elementov užívateľského rozhrania a vzťahov medzi nimi. Takáto aplikácia môže byť s plnou funkčnosťou realizovaná v klasickej podobe pre prostredie operačného systému Windows a rovnako aj v podobe určenej pre internetový prehliadač.

Štandardne je súčasťou operačného systému Microsoft Windows 7, Vista a Server 2008, od verzie Service Pack 2 aj Windows XP a Server 2003 [27, 28].



Obrázok č. 30: Prehľad štruktúry pre .NET Framework verzie 3.0 [28]

#### 4.4 PRÍSTUP A MANIPULÁCIA SO VSTUPNÝMI DÁTAMI

SDK popísané v kapitole 4 (časti 4.1.1), dnes už ako súčasť Windows SDK, poskytuje možnosť využiť samotný grafický tablet, samozrejme v nevyhnutnej symbióze so stolovým či prenosným počítačom, pre tvorbu Tablet PC aplikácií i v prostredí napr. operačného systému Microsoft Windows Vista či Microsoft Windows 7. Užívateľ tak získava možnosť spoznať základné črty, ktoré robia Tablet PC takým osobitým, a oboznámiť sa s nimi v dostatočnom rozsahu. Jedná sa pritom o použitie stylusu pre pohyb po obrazovke, ako náhradu myši a klávesnice, či využitie priameho prevodu písaného textu do textového dokumentu a mnohé ďalšie. Dané SDK navyše obsahuje súčasti pre spracovanie dát prenášaných z tabletu priamo pri používaní stylusu.

Windows SDK umožňuje tvorcom aplikácie rozhodnúť sa medzi dvoma programovacími modelmi – WinForms (varianta COM<sup>5</sup> alebo spravovaný kód (managed code)) a WPF.

Pre COM variantu WinForms sú dôležité nasledujúce hlavičkové súbory, týkajúce sa tabletu a dotykovej technológie:

- **msinkaut.h** – obsahuje základné prostriedky pre manipuláciu s digitálnym atramentom (digital ink), reprezentujúcim získaný tok dát,
- **IACOM.h** – obsahuje prvok InkAnalysis pre spracovanie získaného toku dát,
- **RTSCOM.h** – obsahuje triedy ako RealTimeStylus a StylusPlugin pre zber a úpravu toku dát.

Pre WinForms variantu založenú na spravovanom kóde sú podstatné dynamicky spojené knižnice (DLL), zahŕňajúce nasledujúce menné priestory obsahujúce API pre tablet a dotykovú technológiu:

---

<sup>5</sup> Component Object Model, COM, je nízkoúrovňový štandard určený pre softvérové komponenty, navrhnutý spoločnosťou Microsoft v roku 1993, umožňujúci vzájomnú komunikáciu medzi procesmi a tvorbu dynamických objektov pre širokú škálu programovacích jazykov.



- **Microsoft.Ink** – menný priestor obsahujúci API implementujúce samotný digitálny atrament. Zastrešuje všetky základné prvky pre digitálny atrament ako získaný tok dát, jeho rozpoznávanie, vyobrazenie a zachovanie. Dodatočne, počínajúc verziou operačného systému Windows Vista, sú v tomto priestore umiestnené i časti InkAnalysis API, poskytujúce vývojárom možnosť analyzovať digitálny atrament a získať jednotlivé slová, riadky, odstavce apod.,
- **Microsoft.Ink.TextInput** – nový prvok, ktorý sa objavil s príchodom verzie Windows Vista, obsahujúci API využívané komponentom Tablet PC Input Panel, slúžiacim pre rýchlu a intuitívnu prácu s tabletom okamžite po jeho pripojení,
- **Microsoft.StylusInput** – tento menný priestor obsahuje API, schopné manipulovať s dátami prenášanými z tabletu v reálnom čase, známe pod označením RealTimeStylus API,
- **Microsoft.StylusInput.PluginData** – menný priestor spravujúci dáta predávané pluginom určeným pre RealTimeStylus.

WPF poskytuje základy pre tvorbu aplikácií, poskytujúc pritom možnosť kombinácie užívateľského rozhrania, dokumentov, mediálneho obsahu apod., za využitia celého dostupného výkonu počítača. Pre prístup k dátam z tabletu je možné použiť nasledujúce menné priestory:

- **System.Windows.Controls** – využíva InkCanvas, prvok pre získanie a zobrazenie digitálneho atramentu, teda vstupných dát, a následnú manipuláciu s nimi. V prípade potreby prostého vyobrazenia digitálneho atramentu je vhodné použiť prvok označený ako InkPresenter,
- **System.Windows.Ink** – poskytuje triedy pre spracovanie a manipuláciu s digitálnym atramentom,
- **System.Windows.Input** – obsahuje množstvo tried pre prístup k stylusu, súvisujúcich s digitálnym atramentom a potrebnými nastaveniami,

- **System.Windows.Input.StylusPlugIns** – zahrňa triedy priamo pre manipuláciu so vstupnými dátami (jedná sa o obdobu API, používaného pri použití WinForms modelu založenom na spravovanom kóde, označeného ako StylusInput) [29].

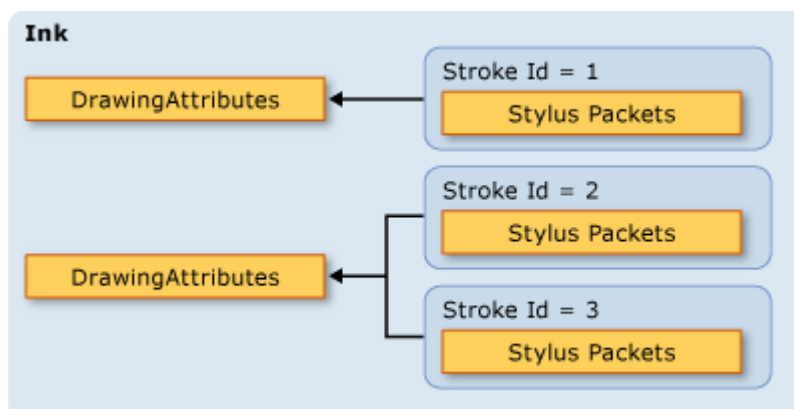
#### 4.4.1 Získanie a reprezentácie vstupných dát

WinForms a WPF majú odlišné modely pre zber, uchovávanie, manipuláciu a vyobrazenie vstupných dát (digitálneho atramentu). Oba modely obsahujú objekty a prvky, umožňujúce aplikácii získať vstupné dáta z tabletu, samozrejme pri použití stylusu, v podobe digitálneho atramentu. WinForms poskytuje prvky ako InkPicture, InkEdit, InkOverlay a InkCollector. InkPicture a InkEdit predstavujú prvky pre zber digitálneho atramentu, ktoré je možné pridať do vyvíjanej aplikácie. InkOverlay a InkCollector sú prvky pre pridanie podpory digitálneho atramentu do už existujúcich aplikácií. WPF obsahuje už spomínané prvky InkCanvas a InkPresenter [29].

U WinForms modelu je pre každý zo spomínaných prvkov (napr. InkPicture, ...) vytvorený objekt triedy Ink (reprezentuje kolekciu tzv. „strokes“<sup>6</sup>, menný priestor Microsoft.Ink). Tento objekt obsahuje dáta pre jeden alebo viac objektov triedy Stroke (reprezentuje jeden „stroke“, menný priestor Microsoft.Ink) a zároveň poskytuje metódy a potrebné nastavenia pre správu a manipuláciu týchto objektov. Ink objekt navyše spravuje životnosť jednotlivých „strokes“ (ktoré obsahuje / vlastní) a je zodpovedný za ich tvorbu a odstraňovanie. Každý „stroke“ má vrámci materského Ink objektu svoj unikátny identifikátor Id [30].

---

<sup>6</sup> „stroke“ - súvislá sekvencia vstupných dát realizovaná užívateľom, reprezentujúca súbor dát od kontaktu stylusu s pracovnou plochou až po prerušenie dátového toku a teda následné vzdialenie stylusu od tejto plochy.

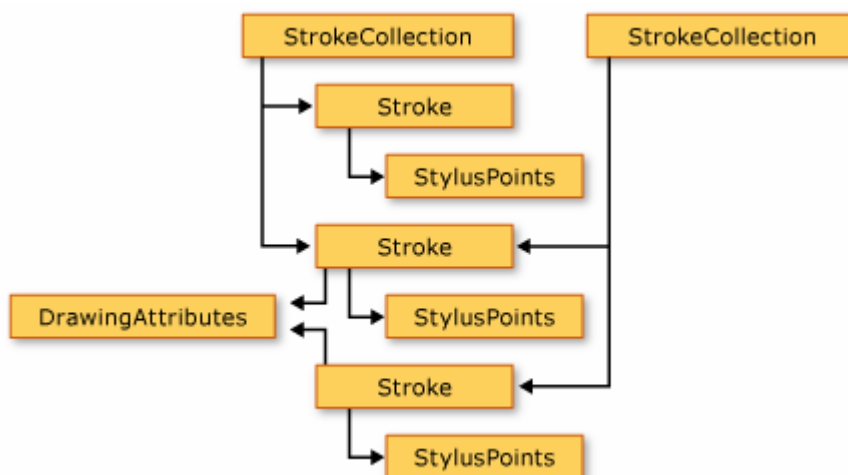


**Obrázok č. 31: Model WinForms.**

Objekt triedy Ink spravuje životnosť jednotlivých objektov Stroke, spolu s paketmi pre príslušný „stroke“, označených unikátnym identifikátorom. Ľubovoľný počet „strokes“ môže odkazovať na rovnaký objekt DrawingAttributes (reprezentuje prívlastky a nastavenie, ktoré sú aplikované v proces zobrazovania digitálneho atramentu, menný priestor Microsoft.Ink) [30]

V prípade modelu WPF spravuje trieda Stroke (reprezentuje jeden „stroke“, menný priestor System.Windows.Ink) svoju životnosť sama za seba. Skupina Stroke objektov môže byť následne zoskupená do tzv. StrokeCollection (reprezentuje kolekciu „strokes“, menný priestor System.Windows.Ink), poskytujúcej metódy pre správu dát ako je testovanie, mazanie či transformácia.

Stroke objekt môže v danom okamžiku patriť do ľubovoľného počtu kolekcí (i do žiadnej). StrokeCollection, ako je zrejmé z daného popisu, tak predstavuje obdobu triedy Ink u modelu WinForms.



Obrázok č. 32: Model WPF.

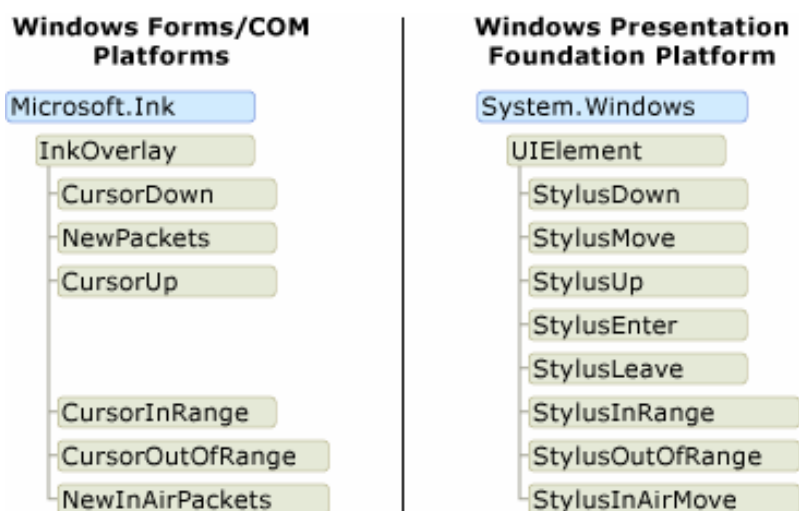
Objekt Stroke je objektom CLR, existujúcim po dobu existencie nejakej referencie naň. Každý Stroke objekt odkazuje na objekt kolekcie StylusPointCollection a prívlastkov DrawingAttributes (menný priestor System.Windows.Ink), ktoré taktiež patria medzi objekty CLR [30]

#### 4.4.2 Správa udalostí

Udalosti vyvolané počas práce s tabletom, teda pri získavaní vstupných dát, sú opäť spravované rozdielnym spôsobom.

U WinForms modelu je daný prvok (napr. InkOverlay, ...) spojený s oknom, prípadne ovládacím prvkom aplikácie a môže byť previazaný s udalosťami vyvolanými pri práci so vstupnými dátami. Výber vlákna, na ktorom sa daná udalosť objaví, závisí na skutočnosti, či bola udalosť vyvolaná pri použití stylusu, myši alebo samotnou aplikáciou.

WPF model obsahuje udalosti pre získavané vstupné dáta v triede UIElement (trieda zodpovedná za vyobrazenie prvkov užívateľského rozhrania, odpovede na dáta získané od užívateľa apod., menný priestor System.Windows), pričom každý ovládací prvok má k dispozícii celú sadu udalostí spojených so stylusom. Daná udalosť sa vždy objaví na vlákne aplikácie.



**Obrázok č. 33: Porovnanie modelu tried pre správu udalostí, ktoré môžu byť vyvolané pri použití stylusu [30]**

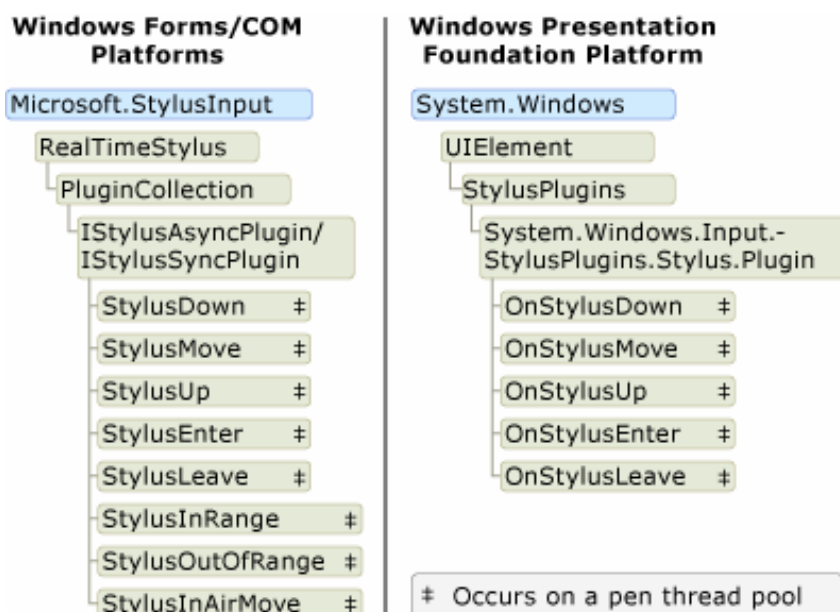
#### 4.4.3 Dáta zo stylusu

Oba modely umožňujú zachytenie a manipuláciu s dátami prichádzajúcimi zo stylusu.

U WinForms modelu sa jedná o vytvorenie objektu triedy `RealTimeStylus`, pripojenie okna alebo ovládacieho prvku k tomuto objektu a následné vytvorenie triedy zavádzajúcej `IStylusSyncPlugin` alebo `IStylusAsyncPlugin`, rozhranie synchrónneho / asynchrónneho pluginu. Požadovaný plugin je následne pridaný do kolekcie pluginov objektu `RealTimeStylus`.

Pre WPF model plní úlohu triedy `RealTimeStylus` trieda `UIElement`. Je zodpovedná za zachytenie dát, vytvorenie triedy dediacej z triedy `StylusPlugIn` a pridanie objektu do kolekcie pluginov `StylusPlugInCollection`.

Bližší popis pre oba modely je možné nájsť v kapitole 4, častiach 4.5 a 4.7.



Obrázok č. 34: Porovnanie modelu tried pre zachytenie dát zo stylusu [30]

#### 4.5 MODEL WINFORMS (VARIANTA SPRAVOVANÝ KÓD)

Model WinForms popisovaný v predchádzajúcej kapitole, ako súčasť SDK dostupného pre operačný systém Windows, a konkrétne varianta založená na spravovanom kóde, obsahuje súčasti pre spracovanie dát prenášaných z tabletu priamo pri používaní stylusu.

Typickým predstaviteľom je trieda označená ako RealTimeStylus, súčasť rozhrania pre programovanie (API) nesúceho názov StylusInput a poskytujúceho prístup k toku dát zo stylusu. Toto rozhranie umožňuje zber, prerušenie a úpravu toku dát nezávisle od samotného vykresľovania a iných operácií. Je navrhnuté tak, aby:

- poskytovalo prístup k toku dát zo stylusu v reálnom čase,
- sa zvýšila výkonnosť a znížila zaťaženosť vlákna pri používaní objektov ako InkCollector, InkOverlay či InkEdit pre zber a spracovanie sekvencií dát.

#### 4.5.1 Architektúra StylusInput API

Primárne je dané rozhranie umiestnené v mennom priestore

Microsoft.StylusInput a Microsoft.StylusInput.PluginData. Aj napriek tomu však dochádza k volaniu určitých tried z assembly označenej ako Microsoft.Ink (jedná sa napr. o triedu Tablet, pre získanie informácií o grafickom tablete pripojenom k systému, apod.).

Pre použitie triedy RealTimeStylus daného API je potrebné v aplikácii, kde sa ju chystáme využívať, vytvoriť objekt RealTimeStylus a previazať ho s príslušným pluginom. Ten sám o sebe predstavuje registrovaný objekt, teda objekt s možnosťou interakcie s danou aplikáciou, ktorému sú od príslušnej časti grafického tabletu (zabezpečujúcej zber dát z hardvéru) predávané požadované informácie. Samotný plugin a jeho rozhranie je pomerne jednoduché. Predstavuje totiž len akúsi kolekciu metód volaných pri zaznamenaní, prípadne zachytení dátového toku, teda akúsi formu obsluhy udalostí.

#### 4.5.2 Špecifikácia pluginov – synchronný a asynchronný

V mennom priestore Microsoft.StylusInput sú tak k dispozícii dva pluginy – IStylusSyncPlugin a IStylusAsyncPlugin. Jedná sa pritom, ako už napovedá názov o:

- **synchronný plugin** – vhodný pre úlohy, ktoré si vyžadujú prístup k dátovému toku v reálnom čase a sú výpočetne nenáročné, ako napr. filtrovanie paketov; tento plugin je v prípade potreby volaný priamo objektom RealTimeStylus na vlákne s vysokou prioritou,
- **asynchronný plugin** – použiteľný pre úlohy bez potreby prístupu k dátam v reálnom čase, ako napr. tvorba a uchovávanie dátových sekvencií; tento plugin je zvyčajne volaný na vlákne užívateľského rozhrania (UI).

Pluginy pre RealTimeStylus musia implementovať synchronne (IStylusSyncPlugin) alebo asynchronne (IStylusAsyncPlugin) rozhranie, prípadne obe.

Zatiaľ, čo je nevyhnutné v každom prípade zaviesť všetky metódy použitého rozhrania, pri samotnom spracovaní dochádza v plugine k volaniu len tých metód, ktoré sú označené v charakteristike DataInterest (z menného priestoru Microsoft.StylusInput.IStylusSync/AsyncPlugin). Objekt RealTimeStylus kontroluje danú charakteristiku v momente, kedy dochádza k jeho pridaniu do synchronnej alebo asynchronnej kolekcie. Preto je vhodné označiť len tie časti, ktoré sa budú následne skutočne využívať.

Pre ďalšie pokračovanie a následnú aplikáciu bude využívané len rozhranie synchronného pluginu, teda IStylusSyncPlugin.

#### 4.5.3 Dátový tok

V prípade použitia grafického tabletu ako polohovacieho zariadenia, kedy je možné dosiahnuť rýchlosť prenosu dátových objektov (paketov) niekoľkonásobne vyššiu ako u bežných polohovacích zariadení (typicky stovky vzoriek za sekundu), predstavuje dostupný dátový tok bohatý zdroj informácií získaných práve z príslušného hardvéru. Následne dochádza k jeho zachyteniu, spracovaniu a analýze, v tomto prevedení v už spomínanom objekte RealTimeStylus.

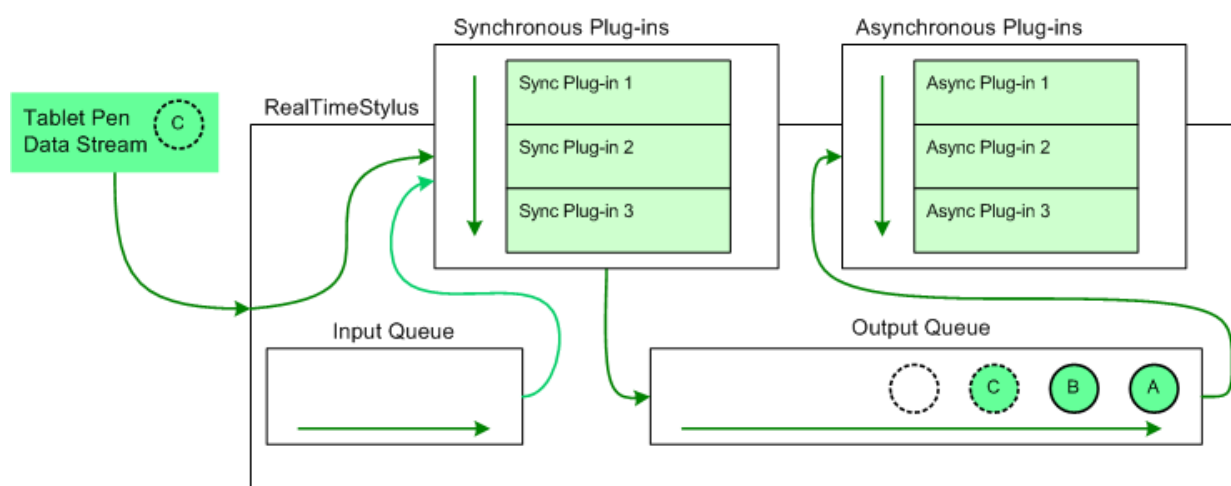
Ten má dve vnútorné fronty pre predávanie dát – vstupnú a výstupnú. So získanými dátami manipuluje nasledovne:

- kontroluje najskôr dátový objekt vo vstupnej fronte a následne v samotnom dátovom toku, ktorý má byť práve spracovávaný; tým je zabezpečené prioritné spracovanie dát a požiadaviek definovaných užívateľom (vo vstupnej fronte), napr. výber či zmazanie požadovaného úseku dát apod.
- pošle jeden dátový objekt do kolekcie synchronných pluginov; každý synchronný plugin môže pridať dáta do vstupnej a taktiež výstupnej fronty
- akonáhle je dátový objekt zaslaný ku všetkým prvkom synchronnej kolekcie a prebehlo jeho spracovanie, je tento umiestnený do výstupnej fronty pre ďalšie použitie/spracovanie asynchronnými pluginmi za účelom uchovania dátových sekvencií apod.
- prebieha čakanie na nasledujúci dátový objekt, ktorý bude spracovávaný



- zatiaľ, čo výstupná fronta objektu RealTimeStylus obsahuje dáta, dochádza k odoslaniu dátového objektu pluginu z výstupnej fronty k objektom v kolekcií asynchronných pluginov; každý asynchronný plugin môže pridať dáta do vstupnej a taktiež výstupnej fronty [31].

Tablet Pen Data Flow



Obrázok č. 35: Znázornenie toku dát cez RealTimeStylus [31]

V diagrame na Obrázku č. 31 predstavujú krúžky, označené písmenami A a B, dátové objekty už pridané do výstupnej fronty objektu RealTimeStylus, avšak ešte nedoslané do asynchrónnej kolekcie. Krúžok označený písmenom C predstavuje práve spracovávaný dátový objekt. Prázdny krúžok reprezentuje pozíciu uloženia dátového objektu, ktorý bude pridaný ako ďalší v poradí.

Výsledkom spracovania dátového toku tak môžu byť údaje o polohe stylusu či tlaku na pracovnú plochu apod. získané v danom okamžiku z charakteristík popisujúcich príslušný dátový objekt – použitie synchronného pluginu, alebo priamo výsledná sekvencia dátových objektov – použitie asynchrónneho pluginu. Možno tak voliť z dvoch prístupov:

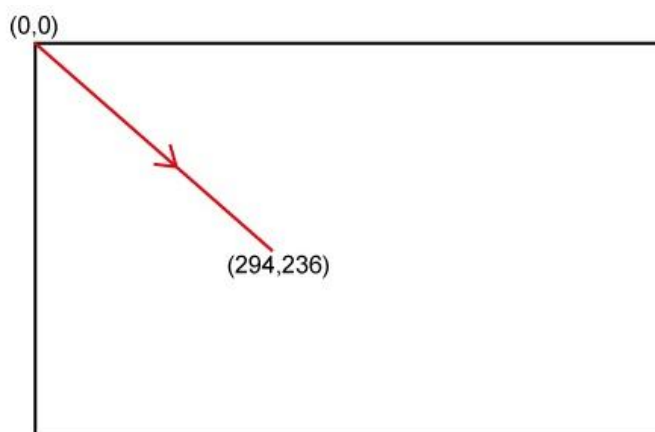
- po prvkoch, pre rýchle okamžité spracovanie a analýzu charakteristík dátových objektov – synchronný plugin,
- po sekvenciách, kedy nie je vyžadovaná analýza jednotlivých dátových objektov a predmetom záujmu sú celé sekvencie - asynchronný plugin.

Ako už bolo spomenuté v kapitole 4, v závere časti 4.5.2, pre ďalšie pokračovanie bude využívaný práve prvý z uvedených, teda synchronný plugin. Je tak možné v každom časovom okamžiku prevádzať súčasne so spracovaním i vizualizáciu získaných dát.

#### 4.5.4 „HIMETRIC“ súradný systém

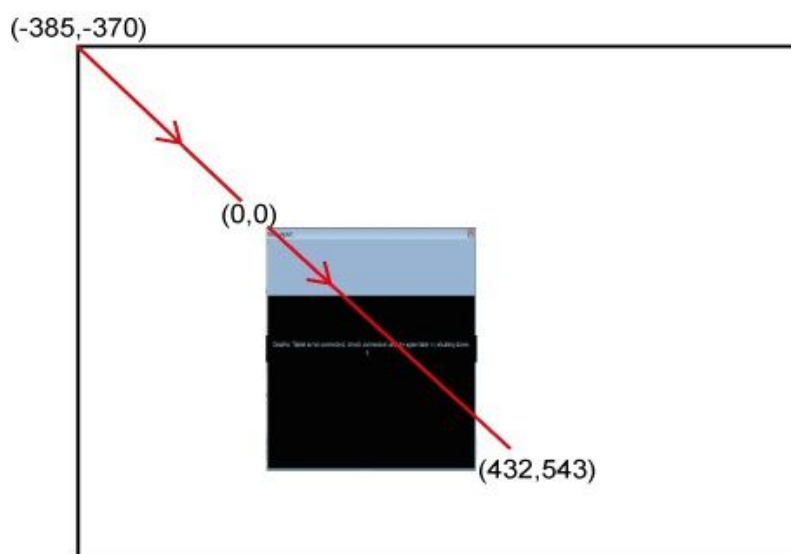
V tejto chvíli je potrebné priblížiť spôsob mapovania polohy stylusu, pri kontakte s pracovnou plochou, na zobrazovacie zariadenie. V prostredí operačného systému Windows je pre polohovacie zariadenia používaný súradný systém označovaný ako „HIMETRIC“.

To znamená, že pohyb z bodu 0 do bodu 1 predstavuje 1 „HIMETRIC“ jednotku, ktorá odpovedá 1/100 mm (rozmery sú odvodené z aktuálneho nastavenia DPI, počtu bodov na palec, zobrazovacieho zariadenia). Súradný priestor je potom reprezentovaný typickým štvorkvadrantovým rozložením s počiatkom v bode (0,0) umiestneným v ľavom hornom rohu zobrazovacieho zariadenia s postupným prírastkom k pravému spodnému rohu (Obrázok č. 36) [4].



Obrázok č. 36: Súradný priestor s počiatkom v bode (0,0)

Situácia sa však mení v okamžiku, kedy začne byť vstup z polohovacieho zariadenia používaný spustenou aplikáciou (využívajúcou .NET Framework), tak ako je tomu v tomto prípade. Je možné sledovať posun počiatku súradného priestoru z ľavého horného rohu zobrazovanej plochy do ľavého horného rohu okna spustenej aplikácie. Zároveň je nutné poznamenať, že smer prírastku zostáva nezmenený a v smere opačnom nadobúdajú súradnice záporné hodnoty.



**Obrázok č. 37: Súradný priestor modifikovaný spustenou aplikáciou**

Naviac, doteraz sa jednalo len o súradnice v systéme „HIMETRIC“. Pre použitie v aplikáciách a pri klasickom zobrazení v bodoch (pixeloch), na aké sme zvyknutý, je nutné realizovať potrebný prevod.

Táto časť môže pôsobiť značne mätúco, keďže platí, že „HIMETRIC“ jednotka predstavuje 1/100 mm a je vzťahnutá k aktuálnemu počtu bodov na palec (DPI). V končnom dôsledku sa však jedná o jednoduchú operáciu delenia celým číslom 2540, ktoré predstavuje počet „HIMETRIC“ jednotiek v jednom palci. Je tak možné určiť polohu kurzoru (reprezentujúceho polohovacie zariadenie) v pixeloch ako na pracovnej ploche zobrazovacieho zariadenia, tak aj v aktívnom okne aplikácie.

```
iX = (int) (g.DpiX * data[0] / 2540);  
iY = (int) (g.DpiY * data[1] / 2540);
```

, kde iX, iY je poloha kurzoru v pixeloch v ose x a y; DpiX, DpiY je odpovedajúci počet bodov na palec (DPI) pre os x a y daného zobrazovacieho systému; data[0] a data [1] sú súradnice bodov v ose x a y získaného dátového tok; 2540 je konštanta popísaná vyššie.

#### 4.6 MODEL WINFORMS - REALIZÁCIA NAVRHNUTÉHO RIEŠENIA

Pre tvorbu testovacej aplikácie bol zvolený operačný systém Microsoft Windows Vista SP1, umožňujúci okamžité použitie grafického tabletu bez potreby dodatočného pridávania ovládačov, a vývojovým prostredím bolo MS Visual Studio 2008 pre jazyk C# (s využitím .NET framework 2.0, ktorý je taktiež súčasťou uvedeného operačného systému).

Samotná testovacia aplikácia slúži pre demonštráciu implementovania a použitia triedy RealTimeStylus, popísanej v predchádzajúcich častiach, a to za účelom získavania dát z grafického tabletu. Jej funkčné prvky možno zhrnúť do niekoľkých bodov:

- testovanie prítomnosti (pripojenia) grafického tabletu
- získanie základných parametrov pripojeného zariadenia
- zachytávanie a analýza dátového toku a následné získanie požadovaných dát, ako napr. poloha stylusu v rovine povrchu tabletu (teda súradnice X a Y), časová značka doby trvania kontaktu stylusu s pracovnou plochou, tlak stylusu voči pracovnej ploche apod.
- uloženie získaných dát pre nasledujúce spracovanie
- vizualizácia získaných dát, výpis získaných súradníc, tlaku apod.

V prostředí MS Visual Studio 2008 bol vytvorený nový projekt, konkrétne Windows Forms Application, a pridané potrebné referencie. V tomto prípade sa jedná o assembly označenú ako **Microsoft.Ink** (súčasť vývojového balíku) a dvoch menných priestorov v nej obsiahnutých – **StylusInput** a **StylusInput.PluginData**. Ďalšou dôležitou súčasťou je použitie menného priestoru **System.Threading** pre podporu viacvláknového programovania.

```
using Microsoft.Ink;  
using Microsoft.StylusInput;  
using Microsoft.StylusInput.PluginData;  
using System.Threading;
```

Následne bol vytvorený objekt **RealTimeStylus**,

```
private RealTimeStylus rts;
```

a v konštruktoze triedy daného projektu pridaný kód potrebný pre jeho použitie...

```
this.rts = new RealTimeStylus(this);
```

Ďalším nevyhnutným krokom je definovanie charakteristík prenášaných paketov, ktoré sú predmetom nášho záujmu, a je možné ho previesť použitím metódy **SetDesiredPacketDescription** (daná charakteristika však musí byť podporovaná použitým zariadením).

```
this.rts.SetDesiredPacketDescription(new Guid[] {  
    PacketProperty.X, PacketProperty.Y,  
    PacketProperty.NormalPressure, PacketProperty.TimerTick,});
```

Poslednou dôležitou časťou, umiestnenou v konštruktoze, je vytvorenie nového pluginu, jeho priadanie do synchronnej kolekcie pluginov a povolenie samotného **RealTimeStylusu**.

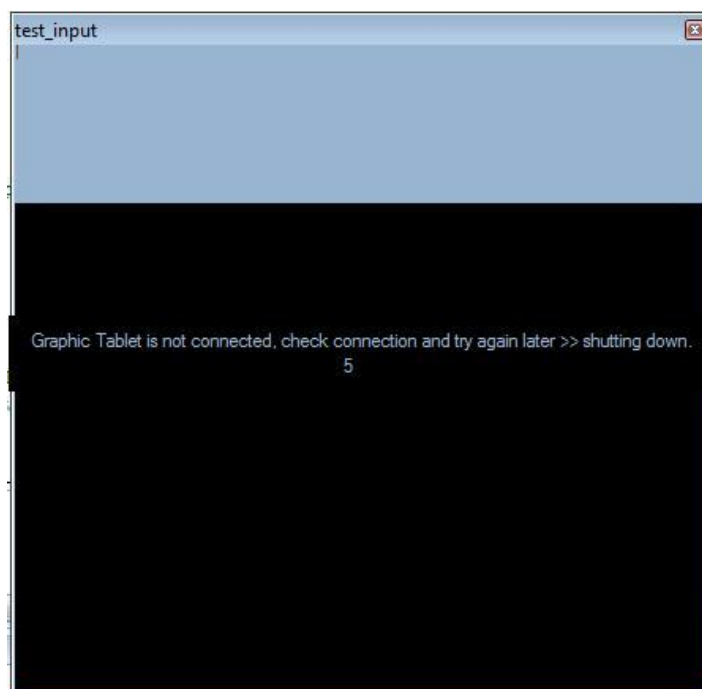
```
StylusData plugIn = new StylusData(this);  
this.rts.SyncPluginCollection.Add(plugin);  
this.rts.Enabled = true;
```

#### *Testovanie prítomnosti (pripojenia) grafického tabletu.*

Pre účely kontroly pripojenia grafického tabletu boli použité metódy, navrhnuté s cieľom získať základné parametre pripojeného zariadenia, popísané v pokračovaní. V tejto časti je nutné rozlíšiť dva stavy, ktoré môžu nastať:

- grafický tablet nie je pripojený už pri štarte aplikácie
- grafický tablet je odpojený za behu aplikácie

V prvom uvedenom prípade, teda pri neprítomnosti grafického tabletu pri štarte aplikácie, dochádza k zobrazeniu výzvy na prevedenie kontroly pripojenia a následne, po uplynutí 5 sekundového intervalu, k ukončeniu aplikácie (Obrázok č. 38).



**Obrázok č. 38: Štart aplikácie bez pripojeného grafického tabletu**

V prípade, že dôjde k odpojeniu, či prípadnému odstráneniu grafického tabletu za behu aplikácie, dochádza k zobrazeniu chybového hlásenia a čaká sa na opätovné pripojenie zariadenia, prípadne ukončenie aplikácie. Pre účely kontroly pripojenia bolo použité samostatné vlákno:

```
new Thread(IsConnecte).Start();
```

#### *Získanie základných parametrov pripojeného zariadenia.*

Pred zahájením samotného zachytávania a analýzou dátového toku je nevyhnutné získať informácie o základných parametroch a podporovaných charakteristikách prenášaných paketov, pripojeného a aktívneho grafického tabletu.

Boli preto vytvorené dve metódy. Prvá, s názvom **ReportOnEachTablet**, slúži pre získanie informácií o základných parametroch pripojeného grafického tabletu. (V prípade použitia prenosného počítača bolo nutné, pomocou vhodnej podmienky, vylúčiť zo zoznamu touchpad, javiaci sa ako k počítaču pripojený grafický tablet.)

Následne po ich získaní, a v prípade že táto vlastnosť je daným zariadením podporovaná, dôjde k vytvoreniu zoznamu všetkých podporovaných charakteristík prenášaných paketov spolu s ich maximálnou, minimálnou hodnotou, rozlíšením a jednotkami, o čo sa stará metóda nazvaná **GetProperty** (na nasledujúcej strane).

```
public string ReportOnEachTablet()
{
    Tablets theTablets = new Tablets();
    string theReport = Environment.NewLine;

    // Iterate over the Tablet objects in the collection
    foreach (Tablet theTablet in theTablets)
    {
        // Exclude laptop touchpad reported as SCREEN
        if (theTablet.PlugAndPlayId != "SCREEN")
        {
            theReport += "Tablet Name: " + theTablet.Name +
            Environment.NewLine;
            .
            .
            .
            if ((theTablet.HardwareCapabilities &
            TabletHardwareCapabilities.Integrated) != 0)
```

```

        theReport += "    Integrated" +
Environment.NewLine;

        // Report on each supported PacketProperty
        theReport += "IsPacketPropertySupported:" +
Environment.NewLine;
        theReport += GetProperty(theTablet,
PacketProperty.AltitudeOrientation, "AltitudeOrientation");
        .
        .
        .
        .
        theReport += GetProperty(theTablet,
PacketProperty.Z, "Z");
        theReport += Environment.NewLine;

    }
    }
    return theReport;
}

public string GetProperty(Tablet theTablet, Guid theGuid, string
name)
{
    string theReport = "";
    // If this particular property is supported,
    // report the name and property metrics information
    if (theTablet.IsPacketPropertySupported(theGuid))
    {
        TabletPropertyMetrics theMetrics =
theTablet.GetPropertyMetrics(theGuid);

        theReport += "    " + name + Environment.NewLine +
            "    Max: " + theMetrics.Maximum.ToString() +
Environment.NewLine +
            "    Min: " + theMetrics.Minimum.ToString() +
Environment.NewLine +
            "    Resolution: " +
theMetrics.Resolution.ToString() + Environment.NewLine +
            "    Units: " + theMetrics.Units.ToString() +
Environment.NewLine;
    }
    return theReport;
}

```



### *Zachytávanie a analýza dátového toku*

V porovnaní s inými polohovacími zariadeniami ako napr. bežná myš, pri ktorej je možné dosiahnuť rýchlosť prenosu paketov maximálne 30 – 35 vzoriek za sekundu, je v prípade grafického tabletu dosahovaná rýchlosť až 200 vzoriek za sekundu (grafický tablet Wacom Intuos 3). To robí z tabletu polohovacie zariadenia, použiteľné rovnako pre bežné úlohy, ako aj pre účely napr. rozpoznávania písma apod.

V danom kroku je nevyhnutné vytvoriť triedu, s ktorou bude plugin (definovaný v konštruktoze) previazaný, a nastaviť jej predka na rozhranie `IStylusSyncPlugin`, reprezentujúce synchronný plugin s možnosťou zaradenia do synchronnej kolekcie.

```
public class StylusData : IStylusSyncPlugin
{
    .
}

```

Táto trieda obsahuje niekoľko definovaných metód, volaných pri zaznamenaní, prípadne zachytení dátového toku. Pre správnu funkčnosť a maximálnu efektivitu aplikácie, ako už bolo spomenuté, je nevyhnutné definovať zoznam tých metód, ktoré budú v aplikácii skutočne použité. Spomínaný zoznam je možné vytvoriť použitím výpočtového typu `DataInterestMask`.

```
public DataInterestMask DataInterest
{
    get
    {
        return DataInterestMask.Packets;
    }
}

```

Ako možno vidieť z uvedeného príkladu, bola vybraná práve jedna z definovaných metód a to metóda **Packets**, keďže predmetom záujmu testovacej aplikácie je zachytenie a následná analýza dátového toku. Je samozrejme možné vybrať i mnohé ďalšie metódy s príslušnými vlastnosťami, napr. `StylusDown`, volanú pri kontakte stylusu s pracovnou plochou grafického tabletu apod.

Následne je možné v tele vybranej metódy/metód prevádzať analýzu dátového toku. Samotný dátový tok obsahuje pakety s príslušnými charakteristikami, definovanými použitím metódy SetDesiredPacketDescription (úvod danej kapitoly). V opačnom prípade, kedy uvedená metóda nie je použitá, je predávaný len defaultný počet charakteristík - typicky 3 – a to poloha v súradniciach x, y a tlak stylusu na pracovnú plochu.

Navrhnutá aplikácia umožňuje získanie celkovo štyroch charakteristík, teda okrem polohy a tlaku na pracovnú plochu i časovú stopu v ms (doba trvania kontaktu stylusu s pracovnou plochou).

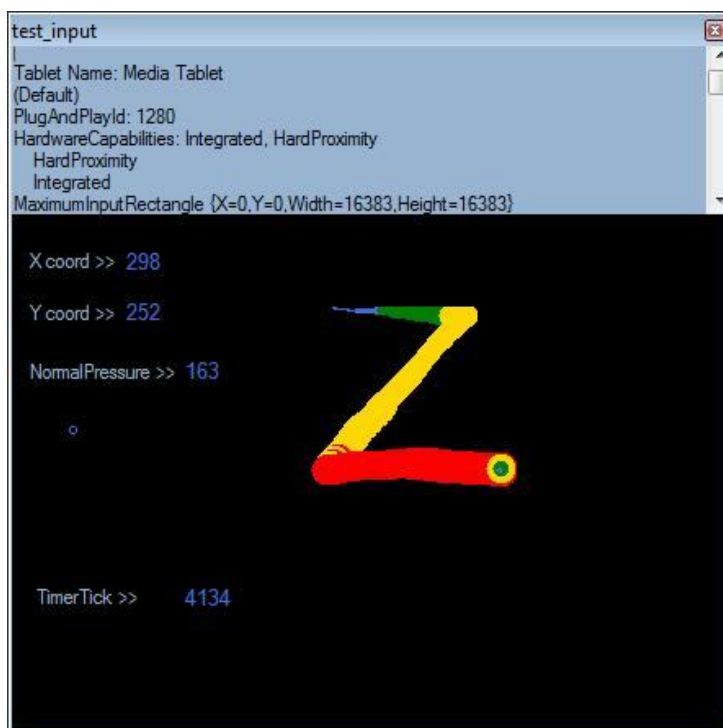
Pre ďalšie spracovanie informácií obsiahnutých v dátovom toku, a predovšetkým polohy, je využívaný súradný systém „HIMETRIC“ (kapitola 4, časť 4.5.4) s cieľom dosiahnuť značnú mieru presnosti pri následnej analýze a spracovaní. Opäť tak grafický tablet získava oproti ďalším polohovacím zariadeniam veľkú výhodu, keďže práca s perom je pre každého jedinca prirodzená a je tak možné dosiahnuť väčšiu citlivosť.

#### *Uloženie získaných dát*

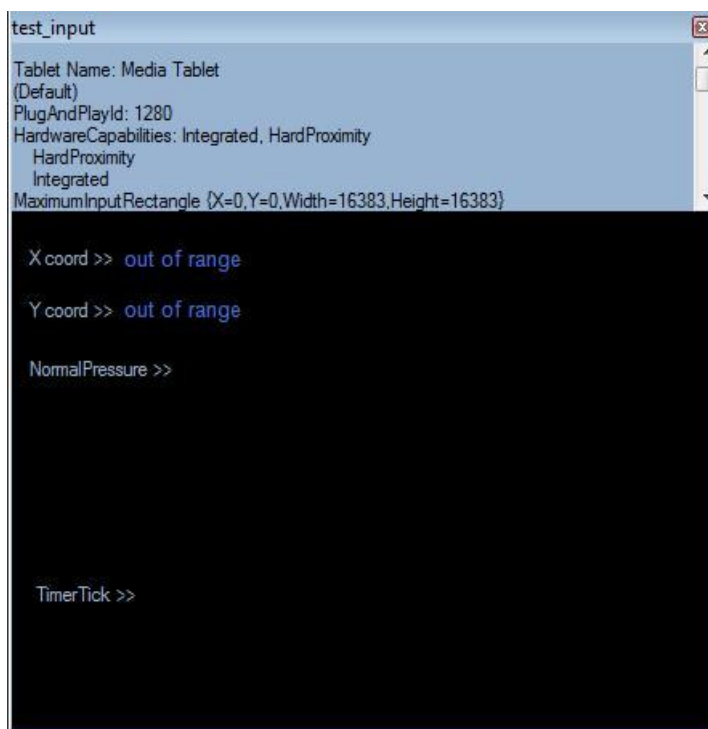
Pre nasledujúce spracovanie a vyhodnocovanie dát je nevyhnutné zabezpečiť, samozrejme pri aktívnom sedení, ich priebežné ukladanie. Je tak následne možné získať informácie o dĺžke trvania daného sedenia, počte súvislých sekvencií realizovaných užívateľom (označovaných ako „strokes“ a reprezentujúcich súbor dát od kontaktu stylusu s pracovnou plochou až po prerušenie dátového toku a teda vzdialenie stylusu od tejto plochy). Ukladanie je prevádzané do textového súboru s unikátnym názvom, vytvoreným na základe dátumu a času začiatku sedenia, či vyprázdnenia definovanej oblasti (zmazanie už zobrazených bodov).

### *Vizualizácia získaných dát*

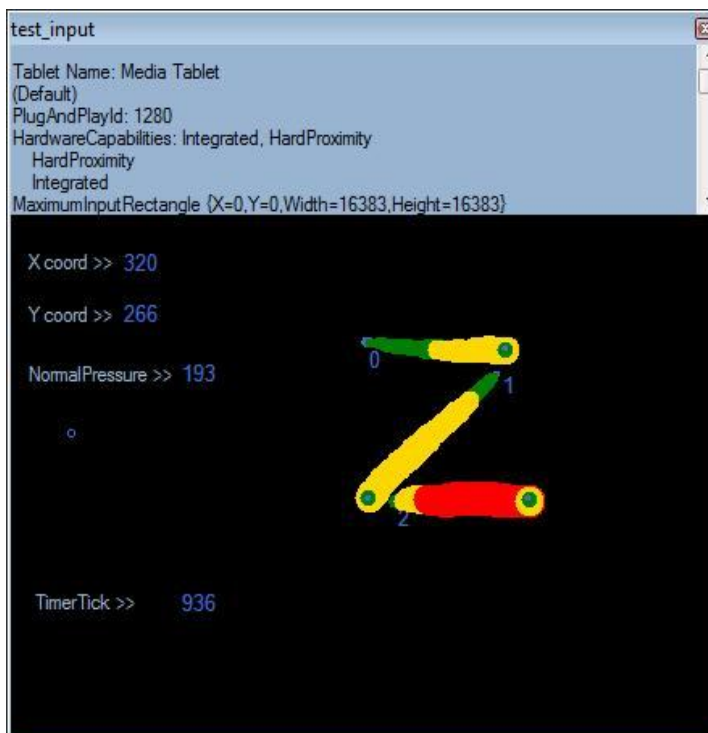
Posledným krokom je vizualizácia získaných dát, teda charakteristík obsiahnutých v každom zo zachytených paketov tvoriacich dátový tok. Užívateľovi je tak poskytnutá možnosť sledovať polohu kurzoru (použitého polohovacieho zariadenia) v pixeloch, a v prípade použitia grafického tabletu i tlak vyvíjaný na jeho pracovnú plochu či časovú stopu v ms (v prípade, že je táto vlastnosť zariadením podporovaná). Obrázky rozhrania navrhutej aplikácie (pre ilustráciu) je možné nájsť v pokračovaní. Prvý z nich predstavuje aplikáciu pri použití stylusu v zvolenej oblasti, druhý pri prekročení definovanej hranice, tretí indexovanie vyvorených „strokes“.



**Obrázok č. 39: Vizualizácia získaných dát na definovanej oblasti**



Obrázok č. 40: Prekročenie hranice definovanej oblasti



Obrázok č. 41: Indexovanie vytvorených „strokes“

## 4.7 MODEL WPF

Druhým modelom popísaným v kapitole 4, časti 4.4, opäť ako súčasť SDK dostupného pre operačný systém Windows, je práve Windows Presentation Foundation (WPF), taktiež obsahujúce súčasti pre spracovanie dát prenášaných z tabletu priamo pri používaní stylusu, poskytujúc pritom možnosť kombinácie užívateľského rozhrania, dokumentov, mediálneho obsahu apod. To vedie k tvorbe prívetivejšieho a príťažlivejšieho užívateľského rozhrania, za súčasného zachovania funkčnosti a efektívnosti poskytovanej modelom WinForms.

Typickým predstaviteľom je trieda označená ako StylusPlugIn, súčasť rozhrania pre programovanie (API) nesúceho názov StylusPlugIns a poskytujúce prístup k toku dát zo stylusu. Toto rozhranie umožňuje zber, prerušenie a úpravu toku dát nezávisle od samotného vykresľovania a iných operácií. Predstavuje tak akýsi vývoj API pre model WinForms, s označením StylusInput.

### 4.7.1 Architektúra StylusPlugIns API

Primárne je dané rozhranie umiestnené v mennom priestore System.Windows.Input.StylusPlugIns. Aj napriek tomu však dochádza k volaniu určitých tried z menného priestoru označeného ako System.Windows.Ink (napr. trieda DrawingAttributes pre špecifikáciu nastavení ako je farba, hrúbka, tvar hrotu stylusu apod. pre daný „stroke“) alebo System.Windows.Input (napr. trieda Tablet apod.).

Úlohu triedy RealTimeStylus, ktorá bola ťažiskovou pre WinForms model, plní v tomto prípade trieda UIElement. Obsahuje vlastnosť označenú ako StylusPlugIns, predstavujúcu kolekciu pluginov (StylusPlugInCollection), do ktorej môže programátor priamo pridať ľubovoľný plugin. Ten sám o sebe predstavuje registrovaný objekt, teda objekt s možnosťou interakcie s danou aplikáciou, ktorému sú od príslušnej časti grafického tabletu (zabezpečujúcej zber dát z hardvéru) predávané požadované informácie v podobe referenčných bodov, tzv. StylusPoints (štandardne obsahujú vlastnosti popisujúce súradnice x a y; podpora ostatných závisí na výrobcovi zariadenia). Samotný plugin a jeho rozhranie je pomerne jednoduché.

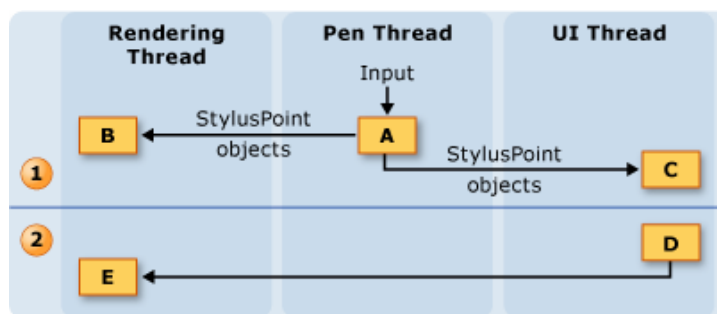
Predstavuje totiž len akúsi kolekciu metód volaných pri zaznamenaní, prípadne zachytení dátového toku, teda akúsi formu obsluhy udalostí. Poradie vloženia pluginov do kolekcie zároveň rozhoduje o poradí prístupu vstupných dát k nim. Napr. ak je pridaný plugin, obmedzujúci vstup do určitej definovanej oblasti, a následne pridaný plugin pre rozpoznávanie gest pri písaní, dostane tento už filtrované dáta od prvého pluginu.

#### 4.7.2 Dátový tok

V prípade použitia grafického tabletu ako polohovacieho zariadenia, kedy je možné dosiahnuť rýchlosť prenosu dátových objektov (paketov) niekoľkonásobne vyššiu ako u bežných polohovacích zariadení (typicky stovky vzoriek za sekundu), predstavuje dostupný dátový tok bohatý zdroj informácií získaných práve z príslušného hardvéru.

Vlákno užívateľského rozhrania (UI), obvyčajne používané pre tento účel v prípade modelu WinForms, nie je dostačujúce pre zber a súčasné vyobrazenie dát, keďže môže dôjsť k jeho zablokovaniu. Ako riešenie priniesla varianta WPF dve dodatočné vlákna, vlákno pre stylus a dynamické vyobrazenie:

- vlákno stylusu – preberá vstupné dáta od stylusu,
- vlákno užívateľského rozhrania (UI) aplikácie – vlákno kontrolujúce užívateľské rozhranie aplikácie,
- vlákno pre dynamické vyobrazenie (rendering) – vlákno zodpovedné za vyobrazenie vstupných dát, v podobe digitálneho atramentu, pri ich zbere.



Obrázok č. 42: Znázornenie modelu využitia vlákien pri zadávaní vstupných dát užívateľom, teda tvorbe „stroke“ [32]

Model na Obrázku č. 42 (na predchádzajúcej strane) je platný pre aplikácie využívajúce prvok InkCanvas (kapitola 4, časť 4.4) a zároveň pre ovládacie prvky vytvorené pre potreby aplikácie. Je možné ho rozdeliť na dve hlavné časti:

1. Udalosti vyskytujúce sa počas tvorby „stroke“ užívateľom

- a. V okamžiku počiatku tvorby „stroke“ užívateľom, začínajú po vlákne pre stylus putovať jednotlivé referenčné body (StylusPoints). Príslušné pluginy, zahŕňajúce tzv. DynamicRenderer (zodpovedný za vyobrazenie digitálneho atramentu, reprezentujúceho vstupné dáta, na povrchu daného prvku; menný priestor System.Windows.Input.StylusPlugIns), tak podľa poradia vloženia do kolekcie dostávajú jednotlivé referenčné body a dochádza k ich modifikácii pred prenosom do prvku InkCanvas, kde dôjde k ich vyobrazeniu.
- b. DynamicRenderer vyobrazuje jednotlivé referenčné body na vlákne pre dynamické vyobrazenie. Tento krok sa deje súčasne s predchádzajúcim.
- c. Prvok InkCanvas obdrží jednotlivé referenčné body na vlákne užívateľského rozhrania aplikácie.

2. Udalosti vyskytujúce sa po ukončení tvorby „stroke“ užívateľom

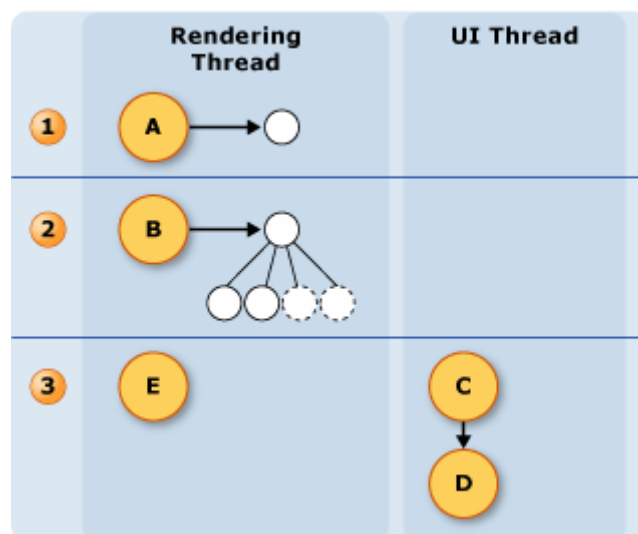
- d. V tomto momente vytvorí prvok InkCanvas objekt Stroke a predá ho prvku InkPresenter, ktorý je zodpovedný za statické vyobrazenie.
- e. Vlákno užívateľského rozhrania aplikácie upozorní prvok DynamicRenderer na skutočnosť, že daný „stroke“ je staticky vyobrazený a tak môže odstrániť jeho vizuálnu reprezentáciu [32].

#### 4.7.3 Vyobrazenie digitálneho atramentu (rendering)

Po dobu tvorby „stroke“ užívateľom zabezpečuje DynamicRenderer vyobrazenie vstupných dát, reprezentovaných digitálnym atramentom, využívajúc pri tom oddelené vlákno, čím je zabezpečený stály a aktuálny tok dát i za situácie, kedy je vlákno užívateľského rozhrania aplikácie zaneprázdnené. DynamicRenderer vytvára pri zbere jednotlivých referenčných bodov, na vlákne pre dynamické vyobrazenie, akýsi vizuálny strom. Po ukončení tvorby „stroke“ užívateľom zažiada DynamicRenderer o upozornenie pri ďalšej akcii vyobrazenia iniciovanej aplikáciou. Po ukončení tejto akcie dôjde k odstráneniu vizuálneho stromu.

Tento proces je znázornený schémou na Obrázku č. 43:

1. Užívateľ začína tvoriť „stroke“
  - A. DynamicRenderer vytvára vizuálny strom
2. Užívateľ pokračuje v tvorbe
  - B. DynamicRenderer pokračuje v budovaní daného stromu
3. Užívateľ končí tvorbu
  - C. prvok InkPresenter pridáva „stroke“ do vizuálneho stromu
  - D. dochádza k statickému vyobrazeniu „stroke“
  - E. DynamicRenderer odstraňuje vizualizáciu

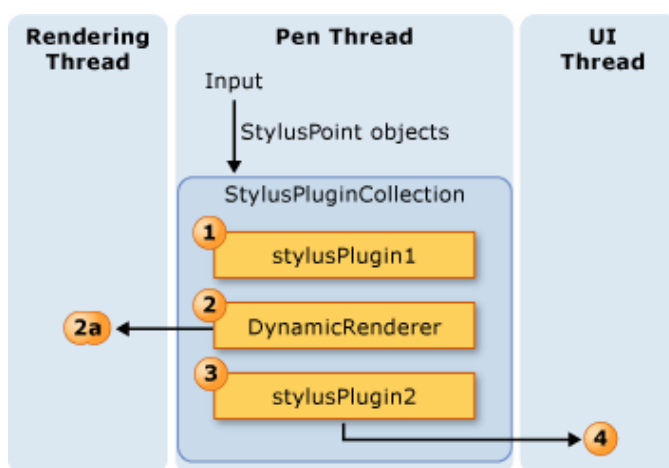


Obrázok č. 43: Znázornenie procesu vyobrazenia digitálneho atramentu [32]



#### 4.7.4 Kolekcia pluginov (StylusPluginCollection)

Ako už bolo spomenuté, jedná sa o kolekciu, do ktorej môže programátor priamo pridať ľubovoľný plugin. Poradie vloženia pluginov do kolekcie zároveň rozhoduje o poradí prístupu vstupných dát k nim. Nasledujúca schéma na Obrázku č. 44 ilustruje možnú situáciu, kedy kolekcia pluginov obsahuje plugin stylusPlugin1, DynamicRenderer a plugin stylusPlugin2 v uvedenom poradí.



Obrázok č. 44: Znázornenie možného usporiadania pluginov v kolekcií [32]

Príklad správanie sa kolekcie:

1. Plugin stylusPlugin1 prevedie úpravu hodnôt pre súradnice x, y daných referenčných bodov.
2. DynamicRenderer obdrží upravené dáta a spracováva ich na vlákne pre dynamické vyobrazenie.
3. Plugin stylusPlugin2 obdrží upravené dáta a ďalej prevádza ich úpravu podľa definície.
4. Aplikácia zoskupuje jednotlivé referenčné body a po ukončení tvorby daného „stroke“ užívateľom zabezpečí jeho statické vyobrazenie.

Predpokladajme, že plugin stylusPlugin1 zabezpečuje obmedzenie vstupných dát, referenčných bodov, len určitej do definovanej oblasti (napr. štvorec) a plugin stylusPlugin2 posúva jednotlivé referenčné body doprava (v smere osi x). Na základe schémy na Obrázku č. 44, na predchádzajúcej strane, počas tvorby „stroke“ užívateľom bude zabezpečené jeho vyobrazenie v medziach definovanej oblasti (plugin číslo 1), ale k jeho posunu (plugin číslo 2) dôjde až po ukončení samotnej tvorby „stroke“, teda až po prerušení kontaktu stylusu s pracovnou plochou [32].

#### **4.7.5 „HIMETRIC“ súradný systém**

V danom modeli, pre použitie v aplikáciách a pri klasickom zobrazení (v pixeloch), už nie je potrebné realizovať prevod z „HIMETRIC“ systému, ako tomu bolo u varianty WinForms (kapitola 4, časť 4.5.4). Tento prevod je zabezpečený samotným rozhraním StylusPlugIns a každý referenčný bod vstupného toku dát je tak priamo popísaný súradnicami x, y v pixeloch.

### **4.8 MODEL WPF - REALIZÁCIA NAVRHNUTÉHO RIEŠENIA**

Daný prístup sa stal východiskovým pre celú problematiku spojenú so získavaním, spracovaním a následným vyhodnotením dosiahnutých výsledkov. Varianta vychádzajúca z modelu WinForms totiž neposkytovala vhodné podmienky prevedenia kalibrácie, následného spracovania dát a dostatočnej interakcie s testovaným subjektom.

Z týchto dôvodov je prístupu založenému na modele WPF venovaná celá nasledujúca kapitola.

## 5. REALIZÁCIA NAVRHNUTÉHO PRÍSTUPU

Daná kapitola sa zameriava na návrh a popis implementácie vhodnej metódy pre zber a spracovanie dát získaných z grafického tabletu, ako aj následné vyhodnotenie získaných výsledkov, vychádzajúc zo základných postupov uvedených v kapitole 4, časti 4.7.

Pre tvorbu aplikácie bol zvolený operačný systém Microsoft Windows 7, umožňujúci okamžité použitie grafického tabletu bez potreby dodatočného pridávania ovládačov, a vývojovým prostredím bolo MS Visual Studio 2010 RC pre jazyk C# (s využitím .NET framework 3.5, ktorý je taktiež súčasťou uvedeného operačného systému).

Samotná aplikácia slúži pre demonštráciu implementovania a použitia častí API pre model WPF, popísanú v predchádzajúcich častiach, a to za účelom získavania dát z grafického tabletu, následného spracovania a vyhodnotenia. Jej funkčné časti možno zhrnúť do niekoľkých bodov podľa významnosti:

### *Vedľajšie časti:*

- testovanie prítomnosti (pripojenia) grafického tabletu pri štarte aplikácie,
- zachytávanie a vizualizácia získaných dát v demonštračnej časti aplikácie.

### *Hlavné časti:*

- prihlásenie do aplikácie a uchovávanie zvolených údajov pre konkrétneho užívateľa,
- prevedenie kalibrácie pre zásobník testovacích vzorov,
- zber dát pri použití navrhnutých vzorov na testovacej množine užívateľov,
- analýza dátového toku, spracovanie získaných dát a uloženie normalizovaných šablón pre nasledujúci proces testovania,
- testovanie na množine užívateľov – overenie identity užívateľa (autentifikácia).

### Microsoft Access

Dnes už označovaný ako Microsoft Office Access, je relačný databázový systém pre správu databáz od spoločnosti Microsoft, kombinujúci Microsoft Jet Database Engine (softvérový prvok zodpovedný v systéme správy za tvorbu, získanie, aktualizáciu a mazanie z databázy) s grafickým užívateľským rozhraním a príslušným softvérovým vývojovým balíkom. Je súčasťou celku Microsoft Office od verzie Professional (a vyššie) a bol použitý pre uloženie dát, potrebných k úspešnej realizácii navrhutej metódy, ako aj informácií o užívateľovi.

Dáta sú ukladané vo vlastnom formáte, založenom na Access Jet Database Engine, ktorý sa mení spolu s verziou systému. Napr. v prípade verzie Microsoft Access 2003 je typickou prípona \*.mdb. Podobne ako ostatné aplikácie z balíku Office, i Access je podporovaný objektovo orientovanými programovacími jazykmi ako je Visual Basic for Applications (VBA), C# apod. pri použití rozhrania prístupu k databáze Data Access Object (DAO), ActiveX Data Objects (ADO), či Object Linking and Embedding Database (OLE DB) navrhnutého spoločnosťou Microsoft [33].

### CSML

Alebo aj C# Matrix Library, je kompaktný balík pre lineárnu algebru obsahujúci množstvo operácií s maticami, v podobe aká je známa z prostredia Matlab, Scilab apod. V súčasnosti je dostupná vo verzii 0.9 zo 4. júla 2007 a voľne dostupná na stránke [www.codeproject.com](http://www.codeproject.com). V danej aplikácii bol použitý v časti transformácie súradníc, označenej ako kalibrácia.

### Určenie miery podobnosti

V poslednej časti procesu - testovaní - ktorej výsledok má výpovednú hodnotu ako o efektívite tak aj o celkovej úspešnosti navrhovanej identifikačnej metódy, je nevyhnutné zaoberať sa charakteristikou poskytujúcou informácie o závislosti medzi veličinami.

Mieru štatistickej závislosti medzi premennými určuje korelácia. V štatistike sa jedná o vzájomný vzťah medzi znakmi či veličinami. Korelačný koeficient môže nadobúdať hodnoty v intervale  $[-1; +1]$ .

Korelácia je definovaná len ak je smerodajná odchylka pre obe veličiny konečná a nenulová a hodnota korelácie nemôže prekročiť hodnotu 1 (brané v absolútnej hodnote), čo je dôsledok Cauchy-Schwarzovej nerovnomernosti. [34]

Hodnota korelačného koeficientu  $-1$  značí celkovo nepriamu závislosť, teda čím viac sa zväčšia hodnoty v prvej skupine znakov, tým viac sa zmenšia hodnoty v druhej skupine znakov, napr. vzťah medzi uplynutým a zostávajúcim časom pri absolvovaní určitej trasy na bicykli.

Hodnota korelačného koeficientu  $+1$  značí naopak celkovo priamu závislosť, napr. vzťah medzi rýchlosťou bicyklu a frekvenciou otáčok kolies bicyklu.

V prípade že je korelačný koeficient rovný nule, potom medzi znakmi nie je žiadna štatisticky zistiteľná závislosť.

Matematické vlastnosti korelácie je možné popísať nasledovne:

Korelačný koeficient  $\rho_{X,Y}$  medzi dvoma náhodnými veličinami  $X$  a  $Y$  s aritmetickým priemerom  $\mu_X$  a  $\mu_Y$  a smerodajnou odchylkou  $\sigma_X$  a  $\sigma_Y$  je definovaný ako:

$$\rho_{X,Y} = \frac{\text{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E((X - \mu_X)(Y - \mu_Y))}{\sigma_X \sigma_Y}, \quad (1)$$

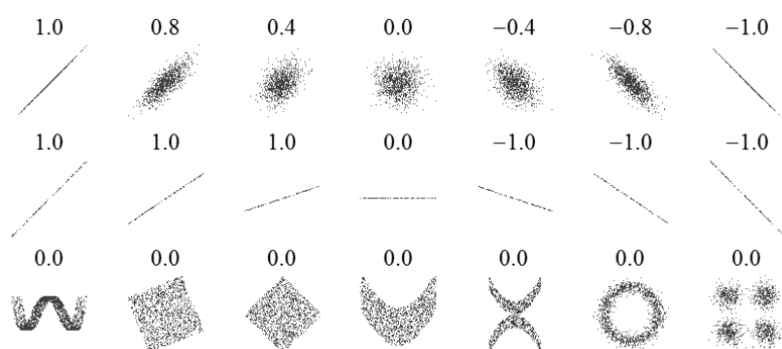
kde čitateľ je tvorený tzv. kovarianciou, ktorá predstavuje strednú hodnotu súčinu odchýliek oboch náhodných veličín  $X$ ,  $Y$  od ich stredných hodnôt. Nadobúda hodnoty z intervalu  $(-\infty; \infty)$ . Poskytuje informáciu o intenzite vzťahu medzi dvoma veličinami. Kovariancia sama o sebe je absolútna veličina a teda pre výpočet relatívnej veličiny, akou je korelačný koeficient, je nutné ju deliť odmocninou násobku rozptylu súboru  $X$  a súboru  $Y$ .

Existuje niekoľko korelačných koeficientov. Najpopulárnejším je takzvaný Pearsonov korelačný koeficient. Charakterizuje mieru lineárnej závislosti medzi premennými. Pre prvky veličiny  $X$  a  $Y$ , značené ako  $x_i$  a  $y_i$  kde  $i = 1, 2, \dots, n$ , je definovaný nasledovne:

$$r_{x,y} = \frac{\sum x_i y_i - n \bar{x} \bar{y}}{(n-1)s_x s_y} = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n \sum x_i^2 - (\sum x_i)^2} \sqrt{n \sum y_i^2 - (\sum y_i)^2}}, \quad (2)$$

$$r_{x,y} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y}, \quad (3)$$

kde  $\bar{x}$  a  $\bar{y}$  sú aritmetické priemery veličín  $X$ ,  $Y$ ,  $s_x$  a  $s_y$  sú smerodajné odchylky veličín  $X$  a  $Y$  a suma je prevádzaná pre  $i = 1, 2, \dots, n$ .



**Obrázok č. 45: Hodnoty korelačného koef. pre rôzne situácie [34]**

Na obrázku č. 45 sú znázornené tri skupiny útvarov so súradnicami  $(x, y)$  spolu s korelačným koeficientom pre každý z nich. Je nutné zdôrazniť, že korelácia zahŕňa mieru rušenia (ako dôsledok prítomnosti šumu) a smer lineárnej závislosti (vrchný rad), ale nie natočenie (stredný rad) ani mnohé aspekty nelineárnych závislostí (spodný rad). Vzor presne v strede má sklon 0, ale v tomto prípade je korelačný koeficient nedefinovaný keďže zmena súradnice  $y$  je nulová.

Interpretácia významu získaného korelačného koeficientu je v súčasnosti spracovaná v niekoľkých variantách od rôznych autorov.

Známostou je varianta vytvorená J. Cohenom pre oblasť psychologického výskumu, ktorá je približená v Tabuľke č. 3. Sám autor však zdôrazňuje, že všetky kritériá sú v značnej miere subjektívne a ich dodržiavanie by nemalo byť príliš prísne. Je to dôsledok skutočnosti, že interpretácia koeficientov závisí na súvislostiach a cieľoch [34].

**Tabuľka č. 3: Význam korelačných koeficientov podľa J. Cohena**

Korelácia	Negatívne	Pozitívne
Malá	-0.3 až -0.1	0.1 až 0.3
Stredná	-0.5 až -0.3	0.3 až 0.5
Veľká	-1.0 až -0.5	0.5 až 1.0

## 5.1 VEĽAJŠIE ČASTI APLIKÁCIE

Predstavujú doplnkové časti aplikácie, s cieľom predchádzať nedefinovaným stavom pri štarte a demonštrovať nespočetné množstvo možností použitého modelu WPF pri riešení predloženej problematiky.

V prostredí MS Visual Studio 2010 RC bol vytvorený nový projekt, konkrétne WPF Application, a pridané potrebné referencie. Následne bol pomocou návrhového prostredia vytvorený navigačný panel aplikácie (na Obrázku č. 46 a 47 na nasledujúcej strane).

### 5.1.1 Testovanie prítomnosti (pripojenia) grafického tabletu

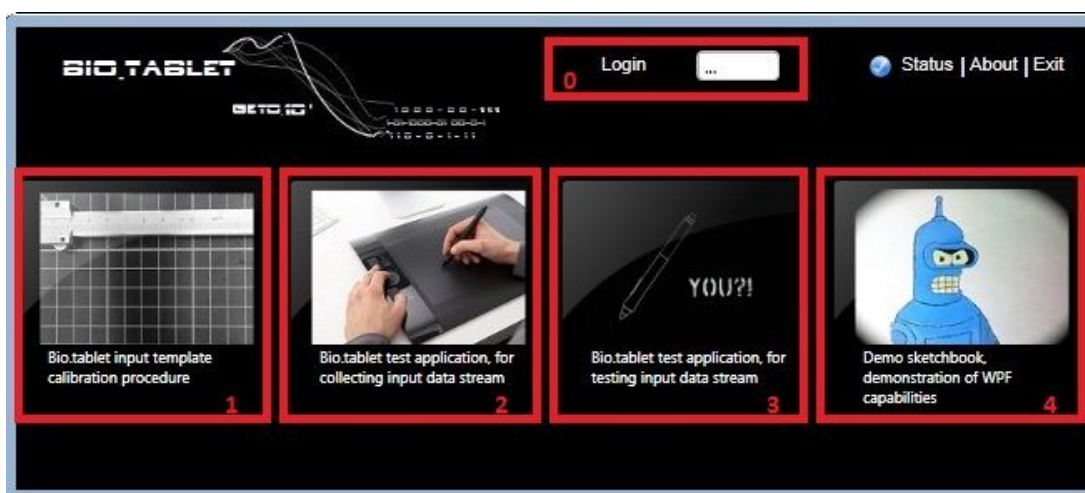
Pre účely kontroly pripojenia grafického tabletu boli použité metódy, navrhnuté s cieľom získať základné parametre pripojeného zariadenia, podobne ako u varianty WinForms (kapitola 4, časť 4.6).

Pri neprítomnosti grafického tabletu počas štartu aplikácie dochádza k zablokovaniu všetkých funkčných prvkov, s výnimkou informačného menu v pravej časti. To slúži pre získanie informácií o stave pripojenia (Status), o aplikácii (About) a taktiež pre ukončenie aplikácie (Exit), (Obrázok č. 46 na nasled. strane).



Obrázok č. 46: Navigačný panel aplikácie – grafický tablet nie je pripojený

Pri úspešnej detekcii pripojenia grafického tabletu počas štartu aplikácie má užívateľ k dispozícii všetky dostupné funkčné prvky, spolu s informačným menu v pravej časti (Obrázok č. 47).



Obrázok č. 47: Navigačný panel aplikácie – grafický tablet je pripojený



Uživatel tak může provést následující činnosti:

- přihlášení do systému (Obrázok č. 47, část 0), potřebné pro sprístupnění procesu kalibrácie a ďalších častí,
- kalibráciu (Obrázok č. 47, část 1),
- po úspešnej kalibrácii spustiť časť aplikácie zameranú na zber dát pri použití navrhnutých vzorov (Obrázok č. 47, část 2),
- v prípade už existujúcich uložených dát vstúpiť do časti simulujúcej proces overenia identity užívateľa pri vstupe do pomyselného systému (Obrázok č. 47, část 3),
- spustiť časť aplikácie demonštrujúcej možnosti WPF pri práci s grafickým tabletom (Obrázok č. 47, část 4).

### 5.1.2 Demonštračná časť aplikácie

Bola vytvorená za účelom priblížiť možnosti WPF, ako varianty pre tvorbu aplikácií, poskytujúc pritom možnosť kombinácie užívateľského rozhrania, dokumentov, mediálneho obsahu apod. pri použití grafického tabletu.

Jedná sa pritom o zachytávanie (prípadnú analýzu či úpravu) a následnú vizualizáciu dátového toku. V danom kroku je nevyhnutné vytvoriť triedu, reprezentujúcu nový plugin, a nastaviť jej predka na triedu StylusPlugin. Tým je zabezpečený príchod toku dát zo stylusu a následne možná modifikácia jednotlivých referenčných bodov, ktoré ho tvoria.

```
class FilterPlugin : StylusPlugin
{
    public FilterPlugin()
    {
    }
}
```

Táto trieda obsahuje niekoľko definovaných metód, volaných pri zaznamenaní, prípadne zachytení dátového toku. Následne je možné v tele vybranej metódy/metód prevádzať analýzu dátového toku na samostatnom vlákne (vlákno pre stylus).

Príkladom môže byť metóda `OnStylusDown`, volaná pri zaznamenaní kontaktu stylusu s pracovnou plochou grafického tabletu.

```
protected override void OnStylusDown(RawStylusInput
rawStylusInput)
{
    base.OnStylusDown(rawStylusInput);
}
```

Najjednoduchším spôsobom pre použitie novovytvoreného pluginu je vytvoriť triedu, reprezentujúcu plochu pre získanie, vizualizáciu dát, prípadne manipuláciu s nimi, a nastaviť jej predka na triedu `InkCanvas`. Zároveň je dôležité pridať plugin do kolekcie pluginov, teda do vlastnosti `StylusPlugIns` vytvorenej triedy. Pre zabezpečenie súčasného vyobrazovania a filtrovania toku dát je navyše potrebné, pri pridávaní do kolekcie pluginov, vložiť navrhnutý plugin pred prvok `DynamicRenderer` (kapitola 4, časť 4.7.4).

```
public class DynamicallyFilteredInkCanvasIn : InkCanvas
{
    FilterPlugin filter = new FilterPlugin();

    public DynamicallyFilteredInkCanvasIn()
        : base()
    {
        int dynamicRenderIndex =
            this.StylusPlugIns.IndexOf(this.DynamicRenderer);

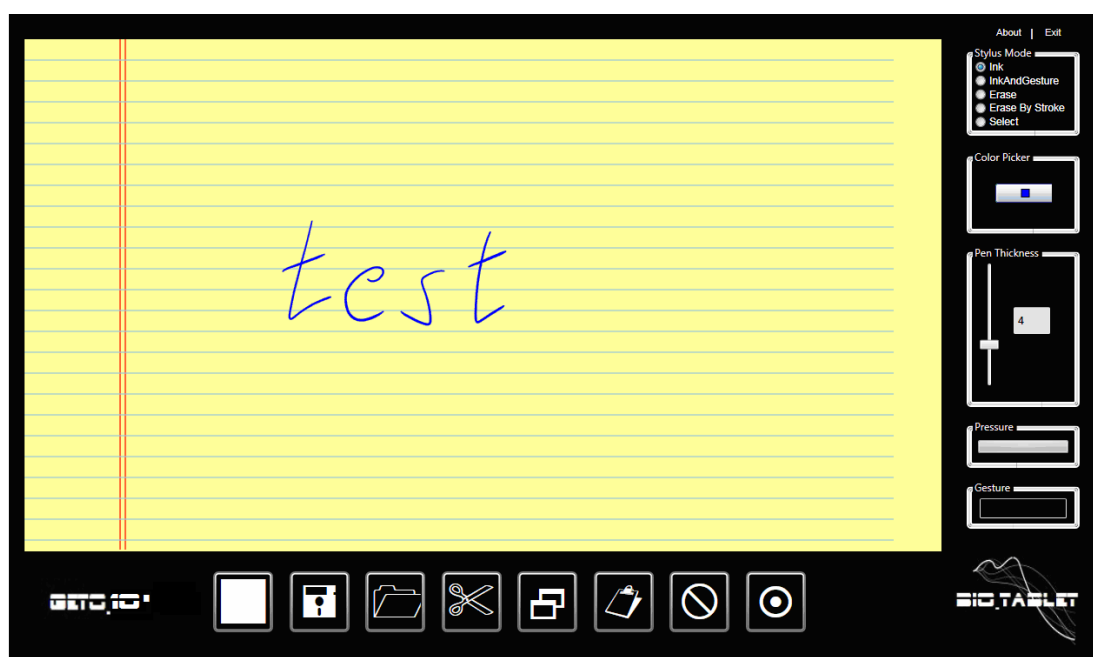
        this.StylusPlugIns.Insert(dynamicRenderIndex, filter);
    }
}
```

Sú tak demoštrované nasledovné možnosti práce s digitálnym atramentom:

- voľba medzi módom pre digitálny atrament
  - len zobrazovanie
  - zobrazovanie a analýza gest
  - mazanie
  - mazanie po jednotlivých „stroke“
  - označenie určitej oblasti

- výber farby pre digitálny atrament,
- výber hrúbky digitálneho atramentu,
- zobrazenie tlaku stylusu na pracovnú plochu grafického tabletu na farebnej stupnici,
- zobrazenie rozpoznaného gesta v príslušnom móde, spoločne so zvukovou podporou,
- prvky pre vyčistenie pracovnej plochy, uloženie jej obsahu, načítanie záznamu z disku, vystrihnutie, kopírovanie, vloženie, odstránenie vybranej oblasti, či označenie celej plochy (všetkých „strokes“).

Náhľad okna demonštračnej časti aplikácie je na Obrázku č. 48.

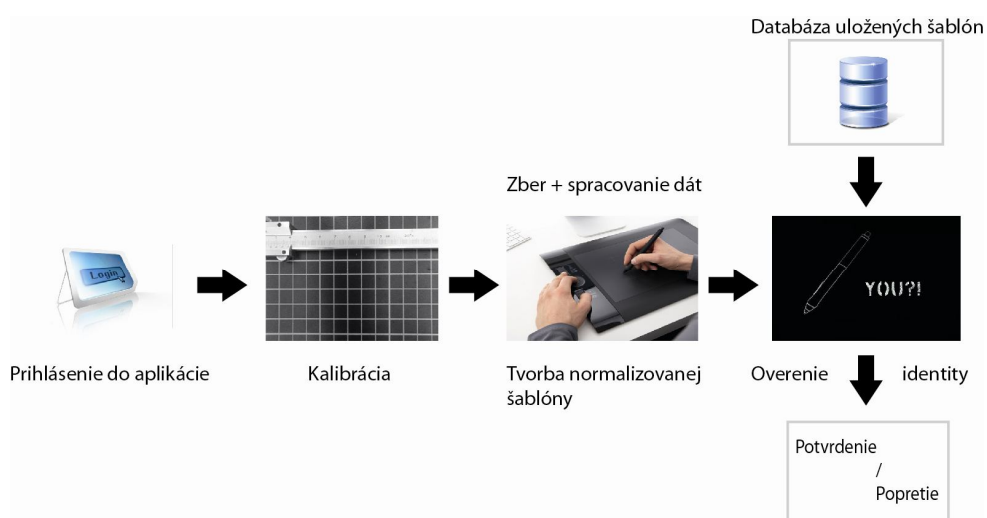


Obrázok č. 48: Okno demonštračnej časti aplikácie

## 5.2 HLAVNÉ ČASTI APLIKÁCIE

Predstavujú ťažiskové časti aplikácie, s cieľom zabezpečiť bezproblémový zber dát, spracovanie, uloženie normalizovaných šablón a následné testovanie pri vhodnej interakcii s užívateľom. Spolu s vedľajšími časťami (kapitola 5, časť 5.1) tak vytvárajú komplexnú aplikáciu, umožňujúcu realizovať jednotlivé kroky navrhutej metódy. Pre dané potreby je vhodné rozdelenie procesu do 4 významných krokov:

1. Prihlásenie do aplikácie a uchovávanie zvolených údajov.
2. Kalibrácia - odstránenie nepresností zaznamenaných pri aplikácii zásobníku s navrhnutými vzormi (viac v kapitole 5, časti 5.2.6) na pracovnú plochu grafického tabletu.
3. Zber dát pri použití navrhnutých vzorov, spracovanie získaných dát a uloženie normalizovaných výsledných šablón pre nasledujúci proces testovania.
4. Testovanie - na množine užívateľov - overenie identity užívateľa.



Obrázok č. 49: Diagram prevedenia algoritmu navrhutej metódy

### 5.2.1 Návrh vhodných typov vstupných dát

Daná kapitola je venovaná rozboru návrhu vhodných typov vstupných dát. Popisuje možnosti interakcie s užívateľom pri získavaní dát z grafického tabletu, návrh vhodných testovacích vzorov a tvorbu použiteľného zásobníku testovacích vzorov ako finálneho produktu danej fázy.

#### Možnosti interakcie s užívateľom

V snahe čo najviac sa vo fáze zberu dát priblížiť podmienkam podobným prípadu použitia klasického pera (pre každého človeka prirodzená činnosť osvojená v priebehu života), je pre vytvorenie vhodných predpokladov úspešnej interakcie užívateľa s grafickým tabletom možné voliť dva prístupy:

- prvý je založený na vykresľovaní vzorov priamo na definovanú pracovnú plochu aplikácie na obrazovke. Užívateľ je navedený k obkresleniu zobrazeného vzoru a získané dáta sú ďalej spracovávané,
- druhý vychádza z tvorby vhodných tréningových, a následne i testovacích vzorov, a ich aplikácii na podklad vyhovujúci tomuto účelu a spĺňajúci požiadavky na dostatočnú jemnosť a hrúbku. Následne je tak možné spomínaný podklad umiestniť na grafický tablet a naviesť užívateľa k jeho obkresleniu. Získané dáta je tak možné ďalej spracovať.

Porovnanie oboch spomenutých prístupov bolo prevedené ako pre aplikáciu založenú na modele WinForms, tak aj pre aplikáciu s modelom WPF, ktorá sa stala východiskovým prístupom pre ďalšie pokračovanie. Varianta vychádzajúca z WinForms totiž neposkytovala vhodné podmienky prevedenia kalibrácie a následného spracovania dát. Získané výsledky sa však stali cennými pre jednoznačný výber vhodnej formy interakcie s užívateľom.

#### Testovacia množina

Pre posúdenie vhodnosti uvedených prístupov interakcie užívateľa s grafickým tabletom bola sledovaná testovacia množina 20 subjektov:

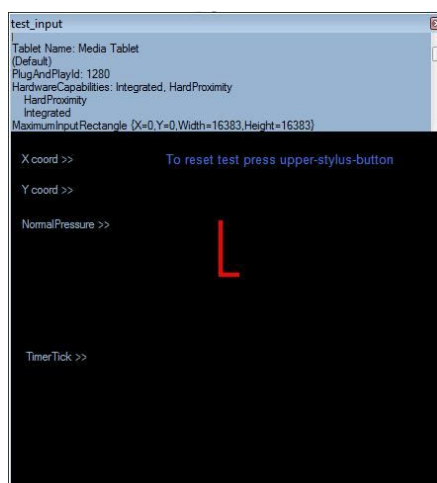
- 13 mužov, 7 žien,
- celkovo 18 pravákov, 2 ľaváci,
- 4 subjekty vo veku do 10 rokov,
- 4 subjekty vo veku od 10 vrátane do 20 rokov,
- 4 subjekty vo veku od 20 vrátane do 30 rokov,
- 4 subjekty vo veku od 30 vrátane do 40 rokov,
- 4 subjekty vo veku od 40 vrátane do 50 rokov
- (jeden z ľavákov spadal do vekovej skupiny od 10 do 20 rokov),
- (druhý z ľavákov spadal do vekovej skupiny od 20 do 30 rokov),

a použité obe varianty aplikácie založené na rozdielnych modeloch (WinForms a WPF).

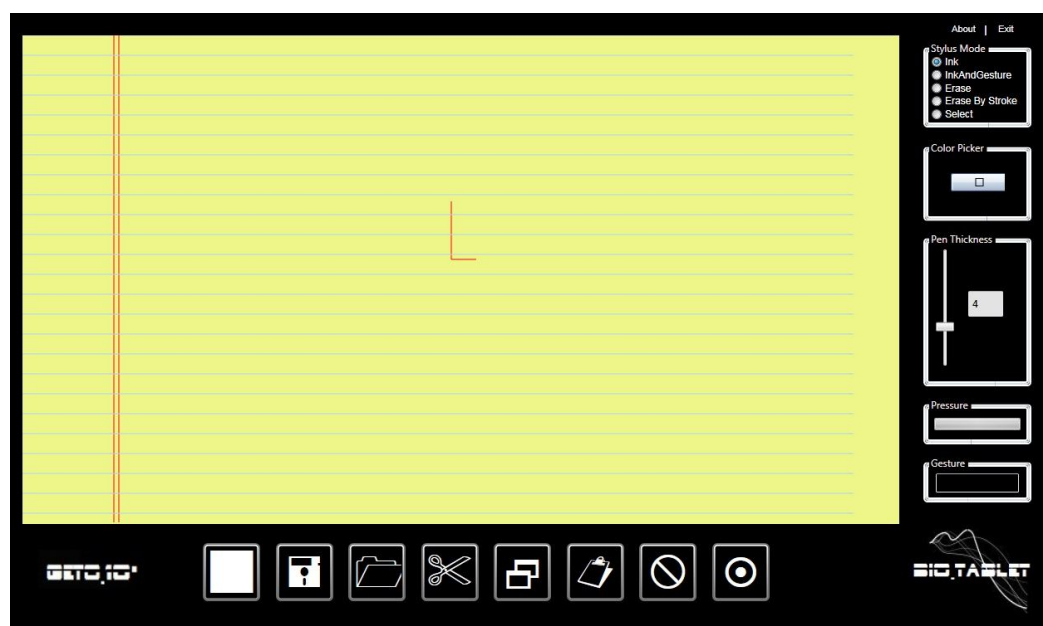
### Implementácia prístupov

Úlohou užívateľa bolo obkresliť písmeno L pri dvoch možných situáciách:

- písmeno bolo zobrazené na pracovnej ploche aplikácie na obrazovke, Obrázok č. 50,
- písmeno bolo vyobrazené na podklade fyzicky umiestnenom na pracovnej ploche grafického tabletu, Obrázok č. 51 na nasledujúcej strane.



Obrázok č. 50: Písmeno L na pracovnej ploche aplikácie (model WinForms)



Obrázok č. 51: Písmeno L na pracovnej ploche aplikácie (model WPF)

Zhodnotenie výberu testovaných subjektov (hodnota v percentách udáva množstvo užívateľov stotožňujúcich sa s daným prístupom):

- varianta s modelom WinForms
  - výber prvého prístupu, s písmenom zobrazeným na pracovnej ploche aplikácie na obrazovke - 10% subjektov
  - výber druhého prístupu, s písmenom zobrazeným na podklade fyzicky umiestnenom na pracovnej ploche grafického tabletu - 90% subjektov vrátane oboch ľavákov
- varianta s modelom WPF
  - výber prvého prístupu, s písmenom zobrazeným na pracovnej ploche aplikácie na obrazovke - 5% subjektov
  - výber druhého prístupu, s písmenom zobrazeným na podklade fyzicky umiestnenom na pracovnej ploche grafického tabletu - 95% subjektov vrátane oboch ľavákov

Významným záverom pozorovania tak bola skutočnosť, že drvivá väčšina testovaných subjektov (s ohľadom na prirodzené držanie pera v pravej či ľavej ruke) potvrdila vhodnosť práve druhého z prístupov, kedy bol sledovaný vzor fyzicky umiestnený na pracovnej ploche grafického tabletu. Daná možnosť sa tak pre testovaného užívateľa javila ako vhodnejšia a prirodzenejšia, čo bolo použité ako východisko pri ďalšom pokračovaní s modelom WPF.

### Návrh vhodných testovacích vzorov

Samotnému procesu návrhu predchádzala konzultácia s grafologičkou. Grafológia, ako forma psychologickéj diagnostiky zaoberajúca sa rukopisom a jeho vzťahom k osobnosti, nie je schopná poskytnúť jednoznačné riešenie danej problematiky. Jedná sa však o jednu z projektívnych metód psychológie, založenej na výrazoch a symboloch, pomocou ktorej je možné diagnostikovať osobnosť. Vychádzajúc tak zo základnej metodiky bola vyvedená potreba návrhu vzorov s určitými charakteristikami, ktorými by sa daný vzor mal vyznačovať. Jednalo sa konkrétne o prítomnosť:

- ostrého uhlu
- tupého uhlu
- pravého uhlu
- hladkého spojitého priebehu

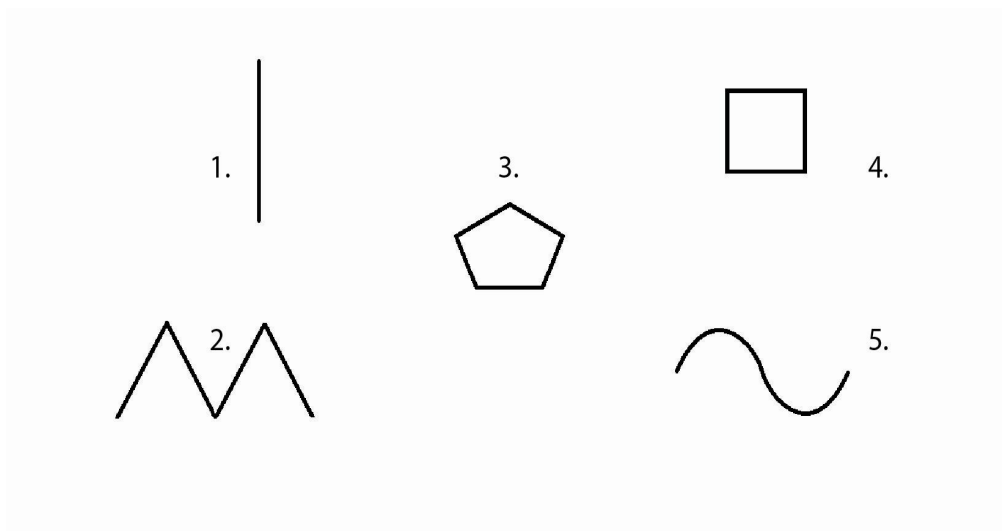


**Obrázok č. 52: Prehľad charakteristík, ktoré boli pri návrhu vzorov predmetom záujmu (zľava ostrý, pravý, tupý uhol a hladký priebeh)**



Ďalšou požiadavkou bola v prípade vzorov s prítomnými uhlami existencia minimálne troch uhlov totožného typu pre konkrétny vzor. Na základe zadaných požiadaviek bolo navrhnutých 5 vzorov tak, aby pri použití celého zásobníku bolo dosiahnuté ich rovnomerné rozmiestnenie na pracovnej ploche grafického tabletu.

Prehľad navrhnutých vzorov (uvedený je vždy pracovný názov):

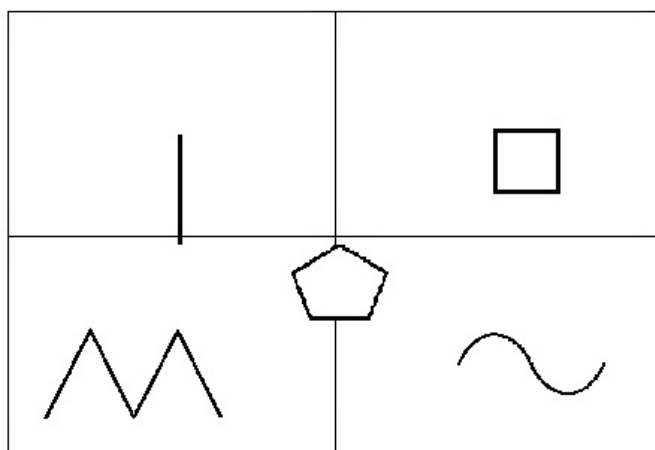


**Obrázok č. 53: Prehľad navrhnutých vzorov**

1. úsečka – nezodpovedá základným požiadavkám, plní však úlohu „štartovacieho“ vzoru, ktorý slúži na oboznámenie užívateľa so systémom a procesom obkresľovania spolu so všetkými náležitosťami. V konečnom dôsledku však nemá pri ďalšom spracovaní výraznejší význam.
2. M – vzor tvaru veľkého tlačeneého písmena M obsahujúci tri ostré uhly.
3. pentagon – 5-uholník, reprezentujúci vzor obsahujúci päť tupých uhlov.
4. štvorec – vzor so štyrmi pravými uhlami.
5. sin – vzor predstavujúci hladký spojitý priebeh, vzhľadovo sa ponášajúci na priebeh goniometrickej funkcie sinus v rozsahu jednej periódy.

### Zásobník testovacích vzorov

Finálnym produktom danej fázy bol súbor navrhnutých vzorov, umiestnený na spoločnom podklade a nesúci označenie zásobník. Ako podklad bol zvolený biely kancelársky papier hmotnosti 80 g/m<sup>2</sup>, vyhovujúci svojou hrúbkou daným potrebám, a vzory boli vytlačené na laserovej tlačiarňi HP LaserJet 1018.



Obrázok č. 54: Finálny zásobník navrhnutých testovacích vzorov

### 5.2.2 Prihlásenie do aplikácie a uchovávanie zvolených údajov

Predstavuje dôležitý krok pred začiatkom aktívneho sedenia užívateľa, teda kalibrácie, následného ďalšieho spracovávania dát a testovania navrhnutých vzorov za účelom overenia identity jedincov.

Pre prihlásenie je potrebné unikátne užívateľské meno (login), pridelené správcom. Záznam všetkých užívateľov, spoločne s uchovávanými údajmi ako dátum / čas prihlásenia apod., je uložený v databáze. Pre tieto účely bola použitá databáza navrhnutá v prostredí Microsoft Access 2003 (kapitola 5, úvod časti 5.2), plne vyhovujúca potrebám aplikácie a lokálneho prístupu k dátam.

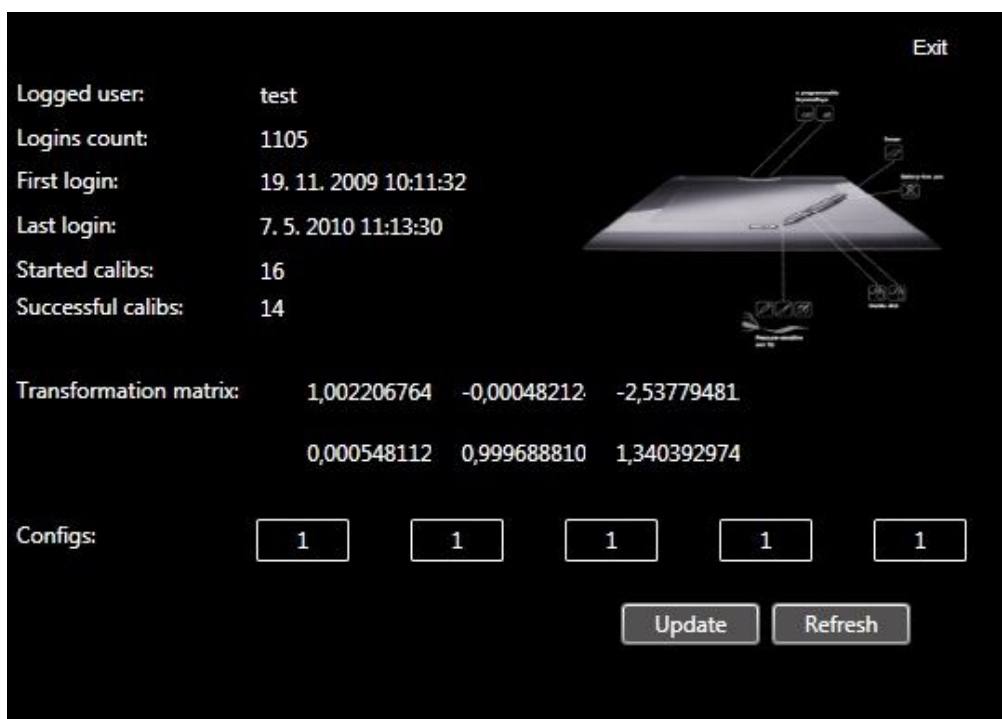
Po prihlásení je navyše možné získať prístup k uchovávaným údajom informačného charakteru ako sú:

- login užívateľa,
- počet prihlásení,
- dátum a čas prvého prihlásenia,
- dátum a čas posledného prihlásenia,

- počet začatých kalibrací,
- počet úspěšně ukončených kalibrací,
- a další údaje bez výpovednej hodnoty pre daný subjekt.

Obdobnou formou by tak bolo možné v prípade potreby uchovávať ľubovoľné údaje o prihlásenom užívateľovi, teda doplnkové osobné informácie ako meno či priezvisko, aktuálnu fotku apod.

Náhľad záznamu uchovávaných údajov pre prihláseného užívateľa je možné nájsť na Obrázku č. 55.



Logged user: test  
 Logins count: 1105  
 First login: 19. 11. 2009 10:11:32  
 Last login: 7. 5. 2010 11:13:30  
 Started calibs: 16  
 Successful calibs: 14

Transformation matrix:
 

1,002206764	-0,00048212	-2,53779481
0,000548112	0,999688810	1,340392974

Configs:
 

1	1	1	1	1
---	---	---	---	---

Update Refresh

**Obrázok č. 55: Náhľad záznamu uchovávaných údajov pre prihláseného užívateľa**

### Implementácia

Pri implementácii bolo použité rozhranie pre programovanie OLE DB, pridaná bola preto referencia na príslušný menný priestor.

```

using System.Data;
using System.Data.OleDb;
    
```

Následne je v prípade potreby možné previesť pripojenie k databáze. Dôležitým krokom je definovanie reťazca pre pripojenie, tvoreného poskytovateľom vytvárajúcim rozhranie pre prístup k dátam samotnou aplikáciou a názvom zdrojovej databázy.

```
string connectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data  
Source=data.mdb";
```

V tomto momente môže prebehnúť samotné pripojenie, získanie hodnôt z databázy použitím SQL výrazu a uloženie do príslušného dátového objektu pre ďalšie použitie.

```
database = new OleDbConnection(connectionString);  
database.Open();  
string queryString = "SELECT ID, Name, Logcount, Firstlog, Lastlog,  
Startedcal, Successfulcal FROM stored";  
loadDataNoClose(queryString);  
database.Close();
```

Obdobným spôsobom, s využitím SQL výrazov, je možné prevádzkať všetky potrebné operácie nad databázou ako tvorba, získanie, aktualizácia a mazanie.

### 5.2.3 Prevedenie kalibrácie pre zásobník testovacích vzorov

Pvým krokom, ktorý vedie k vytvoreniu vhodných predpokladov pre zber dát z grafického tabletu, je zabezpečenie potrebnej kalibrácie, s cieľom odstrániť nepresnosti zaznamenané pri aplikácii zásobníku vzorov na pracovnú plochu grafického tabletu.

Jedná sa teda o postihnutie faktu, že nie je možné vložiť daný zásobník vždy na rovnaké miesto a vzniknutý posun, natočenie apod. je nutné kompenzovať. Daný proces má tak za cieľ zabezpečiť rovnaké podmienky pre každé sedenie užívateľa, získať štandardizovaný výstup a tým prispieť k presnosti výsledného procesu overenia identity na základe definovaných charakteristík.

Vzhľadom na uvedené skutočnosti predstavuje daná časť algoritmu značne kritický krok, keďže na jeho úspešnom prevedení a poskytnutí vhodných výsledkov pre následné spracovanie je závislý celý systém overovania.

Naviac, vychádzajúc z odozvy od zapojených jedincov vo fáze návrhu vhodných typov vstupných dát, požiadavke na samotný proces kalibrácie a rovnako aj nasledujúcich častí musí predchádzať oboznámenie s používaným zariadením a taktiež krátky tréning za účelom osvojenia si jeho ovládania. Pre tieto účely bola použitá demonštračná časť aplikácie, kapitola 5, časť 5.1.2.

### Implementácia

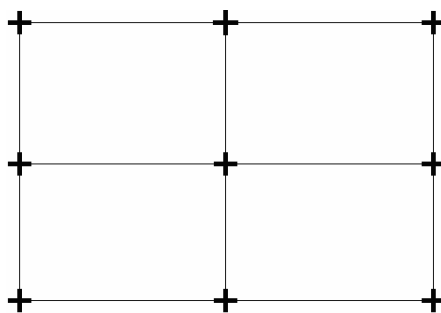
Pre prevod systému súradníc šablóny ( $S_S$ ) do systému súradníc pracovnej plochy grafického tabletu ( $S_T$ ) bol použitý obecný vzťah pre transformáciu:

$$\bar{S}_T = K \cdot \bar{S}_S, \quad (4)$$

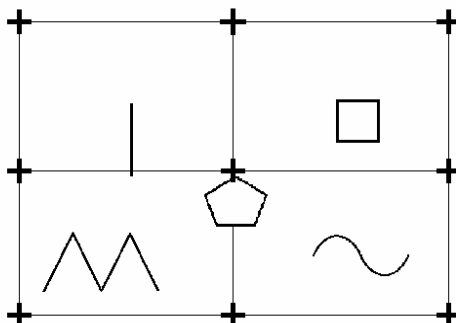
kde  $K$  je matica koeficientov, tzv. transformačná matica, po rozpísaní:

$$\begin{bmatrix} x_T \\ y_T \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \cdot \begin{bmatrix} x_S \\ y_S \\ 1 \end{bmatrix}. \quad (5)$$

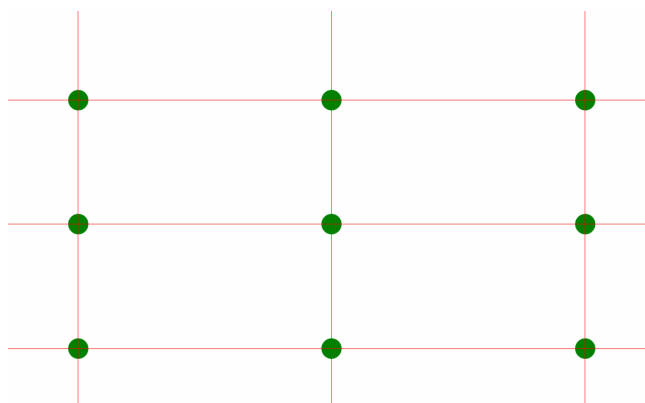
Kalibrácia je proces určenia koeficientov matice  $K$  (transformačnej matice), k čomu je potrebné poznať minimálne štyri dvojice súradníc bodov v rovine šablóny a pracovnej plochy grafického tabletu. V tomto prípade ako kaliber poslúžila pravouhlá sieť bodov umiestnených priamo na šablóne. Pre zlepšenie presnosti je vhodné zvoliť viac ako štyri body, bola preto vytvorená sieť deviatich bodov, a užívateľ bol pri procese kalibrácie vyzvaný k zadaniu („obkresleniu“) každého bodu celkovo trikrát. V konečnom dôsledku tak bolo získaných 27 bodov pre následné spracovanie a získanie transformačnej matice.



Obrázok č. 56: Kaliber použitý v procese kalibrácie



Obrázok č. 57: Zásobník s návrhom testovacích vzorov + kaliber



Obrázok č. 58: Náhľad okna časti aplikácie pre kalibráciu

#### 5.2.4 Zber dát pri použití navrhnutých vzorov a ich následné spracovanie

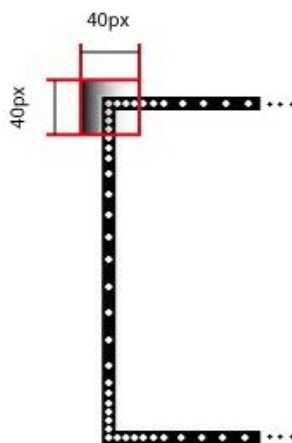
Po prevedení adekvátnej kalibrácie nasleduje ďalší krok v podobe zberu dát pri použití navrhnutého zásobníku vzorov. Jedná sa teda o definovanie prístupu, vedúceho k získaniu normalizovaného výstupu pre príslušný vzor, extrakciu potrebných charakteristík spolu s tvorbou výslednej šablóny a určenie poradia predkladania vzorov počas aktívneho sedenia užívateľa.

### Získanie normalizovaného výstupu

Typickým problémom pri získavaní dát, pri obkresľovaní daného vzoru počas rôznych sedení, je zabezpečenie normalizácie výstupu, ktorý by mohol byť následne použitý pre ďalšie spracovanie. Jedná sa tak nielen o zachovanie unikátnosti prejavu každého jedinca, ale aj o súčasnú kontrolu formy jeho prejavu, teda či v konečnom dôsledku skutočne obkresľuje konkrétny predložený vzor.

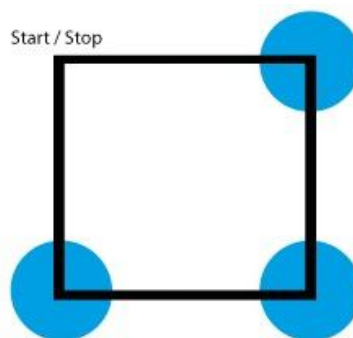
### Implementácia - získanie normalizovaného výstupu

S ohľadom na definované požiadavky bol navrhnutý prístup zastrešujúci udané potreby. Pre každý jeden vzor bola vytvorená štandardná šablóna, vytvárajúca akúsi základnú kostru. Postup spočíval v definovaní jednotlivých bodov, ležiacich na obode vzoru a sústavnej kontrole priechodu týmito bodmi pri obkresľovaní. Tolerancia priechodu daným bodom, zabezpečujúca unikátnosť daného sedenia, predstavovala okolie veľkosti 40x40 pixelov, ktoré dokonale pokrylo potreby použitia na testovacej množine užívateľov.



**Obrázok č. 59: Body definované na obode časti vzoru a vytvárajúce štandardnú šablónu; zvýraznená tolerancia okolia jedného bodu**

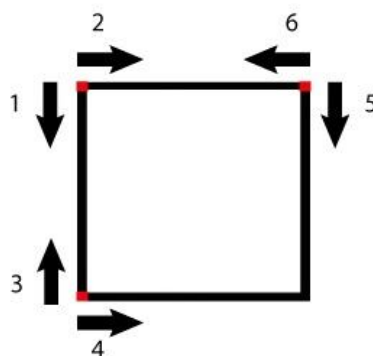
Šablóna je navyše vytváraná s ohľadom na potrebu sledovať predovšetkým oblasti bohaté na informácie, čomu odpovedá hustota rozloženia jednotlivých jej bodov. Pre každý vzor tak boli vybrané príslušné oblasti záujmu a tomu prispôbená už spomínaná hustota.



**Obrázok č. 60: Výber oblastí pre vzor štvorec; 3 oblasti záujmu; posledná slúži ako štartovacia a finalizačná**

Uvedený prístup tak umožňuje, bez ohľadu na aktuálne sedenie, získať pre daný vzor normalizovaný výstup pri súčasnom zachovávaní unikátnosti, ktorá je zabezpečená definovanou mierou voľnosti v podobe tolerancie priechodu každým bodom.

Dôležitým faktom, ktorý nie je možné v danej fáze opomenúť, je rozdielna predstava užívateľov o umiestnení štartovacej / finalizačnej pozície, teda miesta kde užívateľ začne a zároveň skončí pri obkresľovaní konkrétneho vzoru. Pre tento účel bola opäť použitá testovacia množina 20 subjektov (kapitola 5, časť 5.2.1) a na základe získaných informácií vytvorené príslušné varianty šablón pre jednotlivé vzory. Napr. pre vzor štvorec bolo vytvorených šesť variant, Obrázok č. 61.



**Obrázok č. 61: Jednotlivé varianty umiestnenia štartovacej / finalizačnej pozície pre vzor štvorec, spolu so smerom obkresľovania**



### Extrakcia potrebných charakteristík a tvorba výslednej šablóny

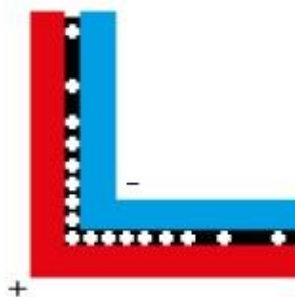
Použitie dostupného grafického tabletu (kapitola 4, časť 4.2) v kombinácii s navrhnutou aplikáciou umožnili získanie nasledujúcich charakteristík:

- aktuálna poloha pera na pracovnej ploche v ose x,y (v pixeloch),
- časová značka pre aktuálnu polohu pera (v ms; merané od počiatku kontaktu s pracovnou plochou),
- tlak vyvinutý užívateľom v danom bode na pracovnú plochu tabletu.

### Implementácia - Extrakcia potrebných charakteristík a tvorba výslednej šablóny

Pre potreby danej aplikácie sa ako nevyhnutnosť ukázala potreba modifikácie prvých dvoch uvedených charakteristík, keďže ich použitie v pôvodnej podobe nebolo vhodné.

Sledovanie presnej polohy pera na pracovnej ploche tabletu bolo nahradené zaznamenávaním diferencie voči štandardnej šablóne a to pri priechoch jednotlivými jej bodmi ležiacimi v definovaných oblastiach záujmu. Priechod vonkajšou oblasťou predstavoval kladnú, vnútornou zápornú diferenciu (Obrázok č. 62), pri maximálnej tolerancii 20 pixelov voči štandardnej šablóne v oboch osách a smeroch.



**Obrázok č. 62: Oblasť kladnej (červená farba) a zápornej (modrá farba) diferencie**

Získanie časovej značky pre aktuálnu polohu pera, vzhľadom k počiatku jeho kontaktu s pracovnou plochou, bolo nahradené rozdielom dvoch po sebe zaznamenaných časových značiek.

Bolo tak dosiahnuté sledovanie časových prírastkov pri prechode medzi jednotlivými bodmi štandardnej šablóny, čo prispelo k posilneniu možnosti sledovať čas reálne strávený užívateľom v aktuálnej oblasti.

V konečnom dôsledku tak bol získaný výstup, normalizovaný pre každý vzor a zároveň unikátny pre každý testovaný subjekt. Následne bolo možné z neho extrahovať dáta pre konkrétne oblasti, ktoré boli predmetom záujmu (napr. pre vzor štvorec tri oblasti, kapitola 5, časť 5.2.4, Obrázok č. 60) a vytvoriť modifikovaný výstup, slúžiaci ako základ pre ďalšie pokračovanie. Modifikovaný výstup tak obsahuje diferenciu pre jednotlivé body oblasti, priemerný tlak pre konkrétnu oblasť a taktiež čas zotrvania v danej oblasti.

Tvorba výslednej šablóny bola spojená s niekoľkými podstatnými faktami a požiadavkami na užívateľa:

- Každý predložený vzor bolo nutné obkresľovať jedným ťahom, len tak bolo možné zaznamenať unikátne charakteristiky pre jednotlivé oblasti.
- Pre každý vzor bolo potrebné získať celkovo desať modifikovaných výstupov s dodržaním určitých pravidiel:
  1. Bola zozbieraná sada piatich výstupov postupne pre všetky vzory.
  2. Nasledovala krátka prestávka po dobu maximálne dvoch minút.
  3. Bola zozbieraná v poradí druhá sada piatich výstupov postupne pre všetky vzory.

Dôvodom uvedených opatrení bola klesajúca pozornosť užívateľov pri prekročení limitu získania piatich výstupov pre konkrétny vzor, čo úzko súviselo so znižovaním ich kvality (miery unikátnosti pre konkrétny subjekt).

- Poradie predkladania jednotlivých vzorov užívateľovi bolo nasledovné: (uvedené sú vždy pracovné názvy) úsečka, M, pentagon, štvorec, sinus... keďže takáto podoba sa, vychádzajúc z postupného návrhu, ukázala čo do kvality (zachovania unikátnych charakteristík) získaných výstupov ako najvhodnejšia.

Úspešné získanie desiatich modifikovaných výstupov tak predstavovalo vhodné predpoklady pre proces tvorby výslednej šablóny podľa navrhnutého postupu:

1. Použitie charakteristiky poskytujúcej informácie o závislosti medzi veličinami, akou je korelácia (bližší popis je možné nájsť v kapitole 5, úvode časti 5.2), pre účely nájdania outliers v množine modifikovaných výstupov (výstupov vykazujúcich malú podobnosť s ostatnými). Na základe získaných informácií boli odstránené vždy štyri výstupy najmenej podobné ostatným, pričom daná hodnota bola získaná ako medián pri sledovaní počtu outliers v priebehu procesu návrhu danej metódy.
2. Zo zostávajúcich šiestich modifikovaných výstupov bol, opäť použitím korelácie, vybratý jeden reprezentant vykazujúci najvyššiu mieru podobnosti s ostatnými, označený za výslednú šablónu a uložený pre použitie vo fáze testovania.
3. Následne boli zo zostávajúcej množiny šiestich modifikovaných výstupov, podobne ako v bode 2, náhodne vybrané a uložené tri výstupy pre potreby fázy testovania, označené ako dočasné šablóny.

#### **5.2.5 Testovanie - overenie identity užívateľa**

Pre fázu testovania boli zvolené dva prístupy:

- offline - pre testovanie boli použité modifikované výstupy (dočasné šablóny) získané počas procesu tvorby výslednej šablóny (kapitola 5, časť 5.2.4),
- online - pre testovanie boli použité modifikované výstupy (dočasné šablóny) získané priamo v danej fáze počas aktívneho sedenia (za prítomnosti testovaného subjektu).

Výsledný proces testovania bol realizovaný ako „dvojstupňový“, v snahe prispieť k zlepšeniu výsledkov dosahovaných pri použití navrhutej metódy. Zvolené označenie možno interpretovať nasledovne:

- Prvý stupeň (začiatok procesu) - užívateľovi je predložený vybraný vzor a získaná odozva v podobe dočasnej šablóny, následne:
  - o pri zhode dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, aspoň 90% je potvrdená jeho identita,
  - o pri zhode dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, aspoň 75% dochádza k prechodu na druhý stupeň,
  - o pri zhode dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, menej ako 75% je identita jedinca popretá.
- Druhý stupeň (ako prechod z prvého) – užívateľovi je predložený vybraný vzor (odlišný od vzoru prvého stupňa) a získaná odozva v podobe dočasnej šablóny, následne:
  - o pri zhode dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, aspoň 90% je potvrdená jeho identita,
  - o pri zhode dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, menej ako 90% je identita jedinca popretá.

### Testovacia množina

Testovacia množina bola v danom prípade tvorená 10 subjektami. Proces získavania dát a tvorby výslednej šablóny totiž vykazoval značnú časovú náročnosť, v rozpätí 25 až 40 minút, došlo tak k redukcii pôvodnej množiny použitej v predchádzajúcich fázach. Popis použitej množiny užívateľov:

- 7 mužov, 3 ženy,
- celkovo 8 pravákov, 2 ľaváci,
- 2 subjekty vo veku do 10 rokov,
- 2 subjekty vo veku od 10 vrátane do 20 rokov,
- 2 subjekty vo veku od 20 vrátane do 30 rokov,
- 2 subjekty vo veku od 30 vrátane do 40 rokov,
- 2 subjekty vo veku od 40 vrátane do 50 rokov,

- (jeden z ľavákov spadal do vekovej skupiny od 10 do 20 rokov),
- (druhý z ľavákov spadal do vekovej skupiny od 20 do 30 rokov).

### **Implementácia - výber vzoru (prvý stupeň)**

Prvým krokom fázy testovania je výber vhodného vzoru a jeho následné predloženie užívateľovi. Algoritmus je navrhnutý tak, aby bol uskutočnený výber vzoru rozdeľujúceho množinu uložených výsledných šablón pokiaľ možno do čo najväčšieho počtu tried a to pri dodržaní adekvátnej rovnomernosti rozloženia. Cieľom tak bolo dosiahnutie stavu, kedy by bolo možné pracovať len s časťou dostupnej testovacej množiny za súčasného zvýšenia efektívnosti a kvality výsledkov daného riešenia.

Sú preto zohľadňované nasledujúce skutočnosti (vzhľadom k dostupnej množine výsledných šablón):

- počet variant šablón, ktoré sa pre konkrétny vzor vyskytujú,
- početnosť jednotlivých zaznamenaných variant šablón pre konkrétny vzor,
- miera odlišnosti výsledných šablón voči štandardnej šablóne pre konkrétny vzor,

a v konečnom dôsledku vybrať vhodný vzor pre predloženie užívateľovi.

Následne bol ako odozva od užívateľa použitý jeden z dostupných testovacích modifikovaných výstupov (offline prístup), alebo aktuálne získaný modifikovaný výstup (online prístup).

### **Porovnávanie**

Proces porovnávania si vyžaduje použitie vhodného postupu, ktorý zabezpečí vyjadrenie miery podobnosti medzi dvoma šablónami. Ten by mal byť schopný poskytnúť určitý rozsah hodnôt pri porovnávaní šablón (dočasnej a uloženej) získaných od toho istého jedinca a takisto pri porovnávaní šablón (dočasnej a uloženej) získaných od rozdielnych jedincov.

Následne je teda možné dospieť k rozhodnutiu, či dané dve šablóny boli /neboli získané od rovnakého jedinca a potvrdiť / poprieť jeho identitu.

Samotný algoritmus porovnávania vychádza zo získaných modifikovaných charakteristík pre definované oblasti záujmu (kapitola 5, časť 5.2.4) – diferencie v jednotlivých bodoch oblasti, priemerného tlaku a času stráveného v oblasti – obsiahnutých vo výsledných šablónach. Zároveň bol navrhnutý hodnotiaci algoritmus, umožňujúci sledovať celkovú zhodu dočasnej šablóny so šablónou uloženou, odpovedajúcou prihlásenému užívateľovi.

Vzhľadom k zvolenému vzoru a záznamom o prihlásenom užívateľovi dochádza pred samotným procesom porovnávania k redukcii testovacej množiny. Na základe varianty šablóny daného vzoru prihláseného užívateľa je vytvorená nová množina užívateľov, u ktorých sa varianta šablóny zhoduje s variantou zaznamenanou. Tým je možné dosiahnuť zvýšenie efektívnosti a presnosti navrhutej metódy, keďže v konečnom dôsledku sa pracuje len s potencionálne zhodnými šablónami a maximálne polovičnou veľkosťou pôvodnej množiny.

### **Implementácia – porovnávanie**

Pre každú oblasť záujmu bol vybratý vektor diferencií a použitím korelácie, ako charakteristiky poskytujúcej informáciu o závislosti medzi veličinami (kapitola 5, úvod časti 5.2), sledovaná zhoda s vektormi diferencií extrahovaných z dostupných uložených šablón. V prípade dosiahnutia maximálnej zhody s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, a miere podobnosti aspoň 0,95 (vyjadrenej hodnotou korelačného koeficientu), zabezpečil hodnotiaci algoritmus pridanie určitého bodového zisku (jeden bod pre oblasť) k celkovej hodnote vypovedajúcej o zhode šablón.

Obdobne bola kontrolovaná zhoda v prípade priemerného tlaku a času stráveného v konkrétnej oblasti. Miesto korelácie bola za účelom kontroly podobnosti jednotlivých šablón použitá diferencia a sledovaná absolútna odchýlka hodnôt dočasnej šablóny od hodnôt dostupných uložených šablón. Pre tlak sa jednalo o hodnotu maximálne 50 úrovní tlaku (z 1024), v prípade času o hodnotu 100 ms. Uvedené hodnoty boli stanovené heuristicky na základe poznatkov nadobudnutých

v priebehu návrhu. V prípade dosiahnutia maximálnej zhody s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, previedol hodnotiaci algoritmus pridanie určitého bodového zisku (jeden bod pre oblasť) k celkovej hodnote vypovedajúcej o zhode šablón. Voľbu daného prístupu si vyžiadal fakt, že získané hodnoty tlaku a času sa pre rôzne sedenia jedinca môžu, i keď častokrát nepatrne, odlišovať a nie je možné voliť formu priameho porovnávania.

V konečnom dôsledku tak hodnotiaci algoritmus umožňuje pre konkrétny prípad zaznamenania 100% zhody dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, dosiahnutie určitej maximálnej bodovej hodnoty, stanovenej pre každý vzor na základe počtu modifikovaných charakteristík a definovaných oblastí záujmu. Následne sa táto hodnota pre fázu testovania stala východiskovým kritériom, umožňujúcim previesť potvrdenie / popretie identity prihláseného jedinca. Napr. pre vzor štvorec predstavuje možná maximálna bodová hodnota číslo 9, ktoré odpovedá 100% zhode šablón v diferencii, priemernom tlaku a čase pre všetky 3 oblasti tohto vzoru.

Naviac, pre jednotlivé udalosti boli nastavené príslušné hranice miery percentuálnej zhody dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, nasledovne:

- potvrdenie identity – aspoň 90% zhoda,
- polovičné potvrdenie identity – aspoň 75% zhoda, prechod na druhý stupeň,
- popretie identity – menej ako 75% zhoda.

Voľba hraníc s pomerne vysokými hodnotami v porovnaní s inými biometrickými metódami, napr. pre rozpoznávanie osôb na základe snímok dúhovky je prah potvrdenia identity asi 66%, prebehla s ohľadom na potrebu testovania navrhutej metódy, pomerne malú testovaciu množinu a tiež značne obmedzenú veľkosť výsledných šablón.

### **Implementácia - výber vzoru (druhý stupeň)**

Vychádza z výsledkov získaných vo fáze porovnávania po prvom stupni výberu vzoru, teda miery percentuálnej zhody dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, vyjadrenej určitou bodovou hodnotou.

Daný stupeň bol implementovaný s cieľom prispieť k presnosti navrhutej metódy, keďže nie je možné pre rozdielne sedenia toho istého užívateľa získať identické výstupy zberu dát – šablóny. Umožňuje tak dodatočne previesť potvrdenie identity, kedy už platia prísnejšie pravidlá s ohľadom na bezpečnosť.

Základom je opäť výber vhodného vzoru a jeho následné predloženie užívateľovi, podobne ako v prvom stupni, pri použití rovnakých mechanizmov a zohľadnení výsledkov z prvej časti. Jedná sa predovšetkým o fakt, že je potrebné vybrať vzor odlišný od vzoru použitého v prvom stupni, no za súčasného dodržania stanovených podmienok.

### **Implementácia – porovnávanie (druhý stupeň)**

Opäť boli použité postupy a východiská ako v prípade porovnávania po prvom stupni výberu vzoru – teda práca s redukovanou testovacou množinou a spracovanie získaných modifikovaných charakteristík pre definované oblasti záujmu, pri použití hodnotiaceho algoritmu.

Ako v predchádzajúcom prípade, i teraz boli pre jednotlivé udalosti nastavené príslušné hranice miery percentuálnej zhody dočasnej šablóny s uloženou šablónou, odpovedajúcou prihlásenému užívateľovi, avšak pri stanovení prísnejších podmienok:

- potvrdenie identity – aspoň 90% zhoda,
- popretie identity – menej ako 90% zhoda.

### **Porovnávanie - výsledky**

Na použitej pracovnej množine bol realizovaný offline i online prístup testovania, oba spomenuté v úvode podkapitoly.



### *Offline přístup*

Bolo prevádzané krížové porovnávanie pre všetkých užívateľov z testovacej množiny, tzn. bola simulovaná snaha o prebratie cudzej identity pri prihlásení pod konkrétnym loginom a následnom predkladaní všetkých dostupných šablón (3 pre každého užívateľa, celkovo 270 pokusov). Pri uvedených podmienkach bolo zistených 25 prípadov zlyhania, kedy bola určená zhoda dočasnej šablóny s neodpovedajúcou výslednou šablónou v databáze, tzv. FAR. FAR alebo false accept rate je pravdepodobnosť, že systém ohlásí úspešnú zhodu medzi vstupnou vzorkou a neodpovedajúcou šablónou v databáze. Udáva tak v percentách pravdepodobnosť nesprávne stanovených zhôd. Pre danú identifikačnú metódu je to 9,3%.

FAR spolu s FRR tvoria dve základné hodnotiace kritériá používané v praxi pre popis úspešnosti biometrickej metódy. V pokračovaní tak bola simulovaná snaha o prihlásenie skutočného vlastníka loginu, teda o potvrdenie identity pri predkladaní všetkých dostupných šablón (3 pokusy pre každého užívateľa, celkovo 30 pokusov). Pri uvedených podmienkach bolo zistených 5 prípadov zlyhania, kedy nebola určená zhoda dočasnej šablóny s odpovedajúcou výslednou šablónou v databáze, teda FRR. FRR alebo false reject rate je pravdepodobnosť, že systém nesprávne deklaruje zlyhanie zhody medzi vstupnou vzorkou s jej odpovedajúcou šablónou z databázy. Udáva v percentách pravdepodobnosť odmietnutia správneho vstupu. Pre danú identifikačnú metódu predstavuje FRR 16,7%.

### *Online prístup*

Daný prístup sa vyznačuje získaním dočasnej šablóny počas aktívneho sedenia užívateľa (za prítomnosti testovaného subjektu) a použitím už existujúcich výsledných šablón. Podobne ako v offline prístupe, i tu bolo prevádzané krížové porovnávanie pre všetkých užívateľov z testovacej množiny pri stanovení rovnakého cieľa. Zistený bol jeden prípad zlyhania, kedy bola určená zhoda dočasnej šablóny s neodpovedajúcou výslednou šablónou v databáze, a už spomínaný FAR tak dosahoval hodnotu blízku 0% (0,3%). Pre ďalšie sledované kritérium, FRR, bola dosiahnutá hodnota 47%.

Pre online prístup bola naviac, pri použití výberovej skupiny z pôvodnej testovacej množiny, zrealizovaná doplnková modifikácie pôvodného prevedenia. Výber bol vzhľadom na charakter úpravy obmedzený len na osoby staršie ako 15 rokov a koncový súbor tak predstavoval 7 jedincov. Spomínaná modifikácia bola zameraná na sledovanie vplyvu fyzickej záťaže v podobe 5 minút strávených na rotopede (pri udržiavaní konštantnej rýchlosti 50 km/h) na hodnoty FAR a FRR. Boli dosiahnuté hodnoty 0% pre FAR a 76% pre FRR.

#### 5.2.6 Zhodnotenie

Pri praktickej realizácii sa objavuje niekoľko problémov spojených s podmienkami prevedenia a navrhnutou metódou.

V prvom rade je to malý počet charakteristík, ktoré je možné pri použití dostupného grafického tabletu získať a následne spracovávať, keďže priemerný tlak na pracovnú plochu a čas strávený v konkrétnej oblasti možno len ťažko považovať za parametre s vysokou mierou unikátnosti a opakovateľnosti. Daný problém by bolo možné vyriešiť použitím niektorého z posledných modelov grafických tabletov, ktoré umožňujú zisk ďalších charakteristík, ako je sklon pera voči pracovnej ploche (uhol udávaný v stupňoch), či natočenie pera okolo svojej osi pri uchopení užívateľom.

Ďalším dôležitým faktom je len obmedzená veľkosť navrhnutých vzorov, čo úzko súvisí s obmedzenou veľkosťou výslednej šablóny. Vzory sú totiž, čo sa týka rozmerov, navrhnuté s ohľadom na potreby vhodnej interakcie aplikácie s užívateľom, v snahe dosiahnuť pri ich obkresľovaní čo najprirodzenejší pohyb pera po pracovnej ploche.

Najdôležitejšou časťou celej metódy je samozrejme fáza testovania. V tomto prípade sú hranice pre potvrdenie / popretie identity, v porovnaní s inými biometrickými metódami, nastavené na značne vysoké hodnoty. Napr. pre rozpoznávanie osôb na základe snímok dúhovky je prah potvrdenia identity asi 66%, v danej aplikácii však voľba prebehla s ohľadom na potrebu testovania navrhnutej metódy, pomerne malú testovaciu množinu a tiež, ako už bolo spomenuté, značne obmedzenú veľkosť výsledných šablón.

Pri realizácii procesu testovania boli sledované dva prístupy, offline a online. Pri prvom z nich, kedy boli použité existujúce dočasné a výsledné šablóny, sa vyskytlo 25 prípadov zlyhania, kedy bola určená zhoda dočasnej šablóny s neodpovedajúcou výslednou šablónou v databáze. Daná situácia je v oblasti biometrických systémov označovaná ako FAR a pre danú identifikačnú metódu a offline prístup predstavuje 9,3%. Je taktiež nutné spomenúť, že asi najpodobnejšia spomedzi dostupných doplnkových biometrických metód – sledovanie dynamiky vlastnoručného podpisu, dosahuje pre FAR hodnotu približne 2%. Jedná sa však o metódu, ktorá od roku 1965 [35] kedy sa objavil prvý systém tohto druhu, prešla nespočetným množstvom testov v rozličných oblastiach bežného života a laboratórnych podmienkach. Priama konfrontácia by tak nebola na mieste. Ďalším dôležitým kritériom pre hodnotenie úspešnosti systému, spolu s uvedeným FAR, je FRR, ktoré zohľadňuje zlyhanie v podobe nevyhodnotenia zhody dočasnej šablóny a jej odpovedajúcej výslednej šablóny z databázy. Udávané je taktiež v percentách a v prípade danej identifikačnej metódy a offline prístupu nadobúda hodnotu 16,7%. V tomto ohľade, pri porovnaní so sledovaním dynamiky vlastnoručného podpisu, kde FRR predstavuje typicky hodnotu blízku 20% [36], sa daná metóda javí ako spoľahlivejšia.

Online prístup, ktorý sa vyznačuje získaním dočasnej šablóny počas aktívneho sedenia užívateľa (za prítomnosti testovaného subjektu) a použitím už existujúcich výsledných šablón, viedol k získaniu odlišných výsledkov. Pre FAR bola dosiahnutá hodnota blízka 0% (0,3%), pre FRR hodnota 47%. Hoci nižšia hodnota FAR je z pohľadu hodnotenia biometrických systémov dôležitejšia ako nižšia hodnota FRR, navrhnutá metóda nie je optimálna. Možnosť prebratia cudzej identity je na základe dosiahnutej hodnoty pre FAR takmer nemožná, vysokú hodnotu pre FRR však možno interpretovať tak, že pri snahe o prihlásenie užívateľa bude takmer polovica pokusov neplatná. Modifikácia daného prístupu, zameraná na sledovanie vplyvu fyzickej záťaž, viedla k získaniu hodnoty blízkej 0% (0,3%) pre FAR a 76% pre FRR. (Online prístup, vrátane modifikácie, bol realizovaný nasledujúci deň po získaní výsledných šablón užívateľov z testovacej množiny.)

Získané hodnoty tak potvrdili fakt, že navrhnutá metóda sa v danom prevedení a s dostupnými prostriedkami vyznačuje pomerne nízkou stálosťou a mierou opakovateľnosti.

Na základe hodnotiacich kritérií pre porovnanie biometrických metód, uvedených v kapitole 2, časti 2.2.3, bol zostavený popis navrhutej metódy odpovedajúci vzoru podľa Tabuľky č. 1. Na základe dostupných faktov je možné podporiť tvrdenie, že navrhnutá metóda v podobe akej bola rozvinutá v danej práci, je schopná len čiastočnej konkurencie iným doplnkovým biometrickým metódam, pričom porovnanie s bežne používanými je vylúčené. Je však na mieste zdôrazniť, že sa jedná len o prvotný návrh.

**Tabuľka č. 4: Porovnanie použitej charakteristiky s najpodobnejšou z doplnkových biometrických charakteristík**

(H=High, M=Medium, L=Low)							
Biometrická charakteristika:	Všestrannosť	Unikátnosť	Stálosť	Zhromažďiteľnosť	„Výkonnosť“	Prijateľnosť	Odolnosť*
Dáta z tabletu	L	L	L	H	L	L	L
Vlastnoručný podpis	L	L	L	H	L	H	L

\* - Odolnosť je označená práve opačnými farbami pretože low je tomto prípade žiaduce namiesto high

Podobne ako v prípade sledovania vlastnoručného podpisu, i daná metóda vyžaduje použitie špeciálneho hardvéru, no veľkým plus sú pomerne nízke náklady potrebné na realizáciu.

Prevádzanie procesu overenia identity, vychádzajúceho z navrhutej identifikačnej metódy, je časovo nenáročné (stovky ms na priemernom PC). Značnou časovou náročnosťou sa vyznačuje len fáza získavania dát pri použití navrhnutých vzorov, kedy sa doba trvania pohybovala v intervale od 25 do 40 minút.

K zvýšeniu úspešnosti navrhutej identifikačnej metódy by navyiac prispelo použitie modernejšieho grafického tabletu. Zároveň by bolo možné pokračovať v zdokonaľovaní návrhu testovacích vzorov s cieľom získania, čo do obsahu informácií, rozsiahlejšej výslednej šablóny a v neposlednom rade i samotného procesu testovania. Bolo by tak možné pokračovať v hľadaní dokonalejších postupov, prispievajúcich k efektívnosti a spresneniu realizácie danej problematiky na rozsiahlejšej množine testovaných jedincov.

## 6. ZÁVER

V tejto práci, ako už bolo spomenuté v úvode, je kladený dôraz na priblíženie problematiky návrhu vhodnej metódy získavania, spracovania a vyhodnotenia dát z tabletu ako takého, so zameraním na možnosť získavať dáta z grafického tabletu použitého v spojení s osobným počítačom.

V priebehu prípravy teoretického úvodu som sa hlbšie zoznámil s biometriou a dostupnými metódami, rovnako ako aj s historickým pozadím vzniku grafického tabletu a vývoja tohoto zariadenia až po súčasnosť, najrozšírenejšími technológiami používanými pri jeho konštrukcii a samotným rozdelením. Získané poznatky sa pre mňa stali nepostrádateľnými pri návrhu a popise implementácie vhodného prístupu pre získavanie dát, založeného na využití dvoch programovacích modelov, tzv. WinForms a WPF, obsiahnutých v softvérovom vývojovom balíku spoločnosti Microsoft, v prostredí operačného systému Windows.

Ďalším krokom bola implementácia navrhnutých metód získavania dát z grafického tabletu pre oba uvedené modely. Programy, vytvorené s využitím dostupných programových prostriedkov v jazyku C#, realizujú problematiku zachytávania a analýzy dátového toku, ukladania získaných dát a následnej vizualizácie. Bola v nich spracovaná celková problematika týkajúca sa získania dát od konkrétného užívateľa a v konečnom dôsledku, na základe poskytovaných možností, prevedený výber prístupu vhodného pre ďalšie pokračovanie, ktorým sa v danom prípade stal prístup založený na modeli WPF.

V práci sa ďalej zaoberám popisom zvoleného prístupu a jeho realizáciou. Jedná sa o priblíženie jednotlivých častí vytvorenej aplikácie – vedľajších a hlavných, použitej testovacej množiny užívateľov a návrh vhodných testovacích vzorov pri tvorbe výsledného zásobníku. Následne bola realizovaná postupná implementácia všetkých hlavných častí, predstavujúcich jednotlivé kroky navrhutej metódy. Aplikácia, vytvorená s využitím dostupných prostriedkov modelu WPF v jazyku C#, rieši celkovú problematiku získania dát pri interakcii s užívateľom a ich použitie pre účely overenia identity. Je v nej spracovaná problematika kalibrácie, s cieľom odstrániť nepresnosti zaznamenané pri aplikácii zásobníku s navrhnutými

vzormi na pracovnú plochu grafického tabletu. Ďalej je to problematika zberu dát pri použití navrhnutých vzorov a taktiež spracovanie získaných dát a uloženie normalizovaných výsledných šablón pre ďalšie použitie. Po úspešnom prevedení popísaných krokov tak nasleduje proces testovania – overenia identity, na upravenej množine užívateľov.

Analýza navrhutej metódy priniesla niekoľko zaujímavých záverov. Kritickou je v danom prípade fáza zberu dát. V prvom rade je, pri použití dostupného grafického tabletu, možné získať a následne spracovávať len obmedzený počet charakteristík, pričom priemerný tlak na pracovnú plochu a čas strávený v konkrétnej oblasti možno len ťažko považovať za parametre s vysokou mierou unikátnosti a opakovateľnosti. Ďalším dôležitým faktom je len obmedzená veľkosť navrhnutých vzorov, čo úzko súvisí s obmedzenou veľkosťou výslednej šablóny. Navyše, časová náročnosť uvedenej fázy, predstavujúca 25 až 40 minút, si pre účely následného testovania vyžiadala redukciu testovacej množiny užívateľov. Proces testovania tak predstavuje posledný a zároveň dôležitý krok pre hodnotenie úspešnosti navrhutej metódy. Tu boli sledované dve hodnotiace kritériá, používané v praxi pre popis úspešnosti daného systému, tzv. FAR a FRR. Pre testovaciu množinu bolo v konečnom dôsledku prevedené overenie identity pre:

- offline prístup s hodnotou 9,3% pre FAR a 16,27% pre FRR,
- online prístup s hodnotou 0,3% pre FAR a 47% pre FRR,
- modifikáciu online prístupu s hodnotou 0,3% pre FAR a 76% pre FRR.

(Bližší popis pre FAR a FRR, ako aj analýzy jednotlivých krokov je možné nájsť v kapitole 5, časti 5.2.5. a 5.2.6)

Na základe uvedených hodnôt pre FAR a FRR je možné konštatovať, že v porovnaní s inými doplnkovými biometrickými metódami, hoci vykazuje takmer identické vlastnosti napr. s prístupom sledujúcim dynamiku vlastnoručného podpisu, neposkytuje navrhnutá identifikačná metóda dostatočne uspokojivé výsledky, pričom porovnanie s bežne používanými metódami je vylúčené. Je však na mieste zdôrazniť, že sa jedná len o prvotný návrh.

## 7. POUŽITÁ LITERATÚRA

- [1] ŠONKA, M.; HLAVÁČ, V. *Počítačové vidění*. Praha: Grada, 1992. ISBN 80-85424-67-3.
- [2] ŽÁRA, J.; BENEŠ, B.; FELKEL, P. *Moderní počítačová grafika*. 1. vyd. Praha: Computer Press, 1998. 448s. ISBN 80-7226-049-9.
- [3] FAUGERAS, O. *Three-Dimensional Computer Vision*. London: MIT Press, 1993. 663s. ISBN-10: 0-262-06158-9.
- [4] JARRET, R.; Su, P. *Building Tablet PC Applications*. Microsoft Press, 2003. 576s. ISBN 0-7356-1723-6.
- [5] CLAYTON E. Crooks. *Developing Tablet PC Applications (Programming Series)*. 1.vyd. Charles River Media, 2003. 350s. ISBN 978-1584502524.
- [6] WAYMAN, J.L.: *Biometric – Now and Then: The development of biometrics over the last 40 years*. H. Daum (ed.) *Biometrics in the Reflection of requirements: Second BSI Symposium on Biometrics 2004*. Bundesamt für Sicherheit in der Informationstechnik und SecuMedia Verlags-GmbH, Bonn/Ingelheim, 2004.
- [7] *An overview of biometrics* [online]. c2010 [cit. 2010-03]. Dostupné z: <<http://biometrics.cse.msu.edu/info.html>>
- [8] *Identity SK spol. s r.o.* [online]. Dostupné z: <<http://www.identity.sk/biometria/ident.htm>>
- [9] *MIT Media Laboratory Vision and Modeling Group - Face Recognition DemoPage* [online]. Posledná úprava 25.7.2002. Dostupné z: <<http://vismod.media.mit.edu/vismod/demos/facerec/>>
- [10] ZVAK, M.: Bachelor's Thesis. Brno: VUT FEKT AMT, 2008. 84s.
- [11] *The Forensic Science Laboratory – DNA Analysis* [online]. c2010. Dostupné z: <<http://www.forensicscience.ie/index.asp?locID=18&docID=-1>>
- [12] *Accuresolution – technology* [online]. Dostupné z: <<http://accuresolution.com/technology.html>>
- [13] *LCS/Telegraphics* [online]. c1995. Dostupné z: <<http://pointing.com/index.html>>.



- [14] *Wacom Europe* [online]. c2010. Dostupné z:  
<<http://www.wacom-europe.com/index2.asp?pid=0•=en>>.
- [15] *Pointing device* [online]. Posledná úprava 2.2.2009 [cit. 2009-02].  
Dostupné z: <[http://en.wikipedia.org/wiki/Pointing\\_device](http://en.wikipedia.org/wiki/Pointing_device)>.
- [16] *Paul Fitt's law* [online]. Posledná úprava 4.1.2009.  
Dostupné z: <[http://en.wikipedia.org/wiki/Fitts'\\_law](http://en.wikipedia.org/wiki/Fitts'_law)>.
- [17] *Prof. Elisha Gray, and his Autotelegraph* [online].  
Dostupné z: <<http://www.hffax.de/history/html/gray.html>>.
- [18] *Historical timeline of Computer Graphics* [online]. Posledná úprava 2.5.2005.  
Dostupné z:  
<[http://sophia.javeriana.edu.co/~ochavarr/computer\\_graphics\\_history/historia](http://sophia.javeriana.edu.co/~ochavarr/computer_graphics_history/historia)>.
- [19] *KoalaPad/Painter* [online]. Posledná úprava 30.12.2008 [cit. 2009-02].  
Dostupné z: <<http://en.wikipedia.org/wiki/KoalaPad/Painter>>.
- [20] *From Apple to Zeta* [online]. Dostupné z:  
<<http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=01673869>>.
- [21] *Graphics tablet – History and background* [online]. [cit. 2009-02].  
Dostupné z: <[http://www.experiencefestival.com/a/Graphics\\_tablet\\_-\\_History\\_and\\_background/id/1439165](http://www.experiencefestival.com/a/Graphics_tablet_-_History_and_background/id/1439165)>.
- [22] *Graphics tablet* [online]. Posledná úprava sep. 2007 [cit. 2009-02]. Dostupné z: <<http://www.nationmaster.com/encyclopedia/Graphics-tablet/screen-hybrid>>.
- [23] ZAHŘÁDKA, J. *Test: Z obyčejného notebooku tablet za dvě minuty* [online]. Posledná úprava 18.5.09. Dostupné z:  
<[http://notebooky.idnes.cz/test-z-obycejneho-notebooku-dotykovy-tablet-za-dve-minuty-paa-/prislusenstvi.asp?c=A090515\\_164511\\_prislusenstvi\\_jza](http://notebooky.idnes.cz/test-z-obycejneho-notebooku-dotykovy-tablet-za-dve-minuty-paa-/prislusenstvi.asp?c=A090515_164511_prislusenstvi_jza)>.
- [24] *.NET Framework* [online]. Posledná úprava 13.1.2010.  
Dostupné z: <[http://en.wikipedia.org/wiki/.NET\\_Framework](http://en.wikipedia.org/wiki/.NET_Framework)>.
- [25] *.NET Framework Developer Center* [online]. c2010.  
Dostupné z: <<http://msdn.microsoft.com/en-gb/netframework/default.aspx>>.
- [26] *Windows Forms (WinForms)* [online]. Posledná úprava 24.12.2009.  
Dostupné z: <[http://en.wikipedia.org/wiki/Windows\\_Forms](http://en.wikipedia.org/wiki/Windows_Forms)>.

- [27] *Introducing Windows Presentation Foundation* [online]. Posledná úprava sep. 2006. Dostupné z: <<http://msdn.microsoft.com/en-us/library/aa663364.aspx>>.
- [28] *Windows presentation Foundation (WPF)* [online]. Posledná úprava 14.1.2010. Dostupné z: <[http://en.wikipedia.org/wiki/Windows\\_Presentation\\_Foundation](http://en.wikipedia.org/wiki/Windows_Presentation_Foundation)>.
- [29] Graff, E. *The Proper Developer Environments for Mobile PC, Tablet PC, and Ultra-Mobile PC Applications* [online]. Posledná úprava 2007 [cit. 2010-01]. Dostupné z: <<http://www.eps-publishing.com/article.aspx?quickid=0704062&page=1>>.
- [30] *The Ink Object Model: Windows Forms and COM versus WPF* [online]. c2010. Dostupné z: <<http://msdn.microsoft.com/en-us/library/ms788719.aspx>>.
- [31] *Architecture of the StylusInput APIs* [online]. Posledná úprava 10.8.2009 [cit. 2009-11]. Dostupné z: <<http://msdn.microsoft.com/en-us/library/ms700688%28VS.85%29.aspx>>.
- [32] *The Ink Threading Model* [online]. c2010. Dostupné z: <<http://msdn.microsoft.com/en-us/library/ms788744.aspx>>.
- [33] *Microsoft Access* [online]. Posledná úprava 15.1.2010. Dostupné z: <[http://en.wikipedia.org/wiki/Microsoft\\_Access](http://en.wikipedia.org/wiki/Microsoft_Access)>.
- [34] *Correlation* [online]. Posledná úprava 8.5.2010 [cit. 2010-05]. Dostupné z: <<http://en.wikipedia.org/wiki/Correlation>>.
- [35] *Dynamic Signature* [online]. Posledná úprava 7.8.2006. Dostupné z: <<http://www.biometrics.gov/Documents/DynamicSig.pdf>>.
- [36] SHEFFIELD, T. *Handwritten Signature Recognition From the Ground Up* [online]. Posledná úprava 3.5.2010. Dostupné z: <<http://www.docstoc.com/docs/27483990/An-Introduction-to-Handwritten-Signature-Recognition-From-the/>>.
- [37] JAIN, A.K.; BOLLE, R.; PANKATI, S. *Biometrics: Personal identification in networked society*. Kluwer Academic Publishers, 2002. 411s. ISBN 0-7923-8345-1.

## **Zoznam príloh**

Príloha 1      CD s programom a elektronickou verziiu tejto práce